FACULDADE DE TECNOLOGIA DE SÃO PAULO

Erick Skorupa Parolin

Data Warehouse Aplicado ao Negócio

São Paulo
2011

FACULDADE DE TECNOLOGIA DE SÃO PAULO

Erick Skorupa Parolin

Data Warehouse Aplicado ao Negócio

Monografia submetida como exigência parcial para a obtenção do Grau de Tecnólogo em Processamento de Dados
Orientador: Prof. Dr**.** Dionísio Gava Junior

São Paulo
2011

AGRADECIMENTOS

Aos meus pais que sempre estiveram presentes nas minhas decisões e dedicaram suas vidas com amor à batalha que é criar.

Ao meu irmão companheiro, com quem sempre compartilho muitos momentos de alegria.

À minha namorada, pela paciência, carinho e compreensão durante esses últimos anos muito atarefados.

Ao meu professor orientador, por conduzir as minhas pesquisas e ajudar no meu processo de desenvolvimento profissional e acadêmico.

RESUMO

O Data Warehouse é uma ferramenta largamente utilizada dentro de grandes organizações como uma estrutura fundamental de integração dos dados que fornece suporte para processos de extração de conhecimento e tomada de decisões desde níveis operacionais a estratégicos. Este trabalho busca primeiramente abordar conceitos técnicos fundamentais sobre a tecnologia, passando por contextos históricos com o intuito de ilustrar a origem do ambiente planejado, e por fim expor aplicações e a funcionalidade da ferramenta como suporte ao negócio. Para tanto, foi aplicado um modelo de pesquisa apoiado em apresentação de alguns exemplos para que seja possível compreender e mensurar a forma como a tecnologia é capaz de transformar o ambiente de negócios.

ABSTRACT

Data Warehouse is a tool widely used in big companies as a essencial structure for data integration which provides support to knowledge extraction and decision-making processes from operational to strategic levels. The purpose of this work is to approach fundamental technical concepts about that technology, including its historical contexts in order to illustrate the origin of the planned environment, and at last show some applications and functionalities of the tool as business support. In order to meet this objective, it was used a research method based on the presentation on some examples for the comprehension and evaluation of the tool as for its performance.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

INTRODUÇÃO

No mundo agressivo e competitivo onde as empresas atuam hoje, mais do que nunca, torna-se necessário a capacidade de reagir rapidamente e se adaptar de acordo com as oscilações e a dinamicidade do mercado que as compete, de modo a garantir perenidade.

Sendo assim, executivos e diretores de grandes companhias necessitam cada vez mais do recurso mais valioso dos tempos contemporâneos: informações. Informações precisas, direcionadas, rápidas e de qualidade, que o permitam tomada de decisões estratégicas contribuindo para que as organizações atinjam suas metas e obtenham vantagens competitivas.

Com os crescentes avanços na área da tecnologia da informação, encontramos um imenso volume de dados provenientes de sistemas de informação dentro das empresas. Porém, para que estes dados espalhados por diversos sistemas legados dentro da organização se tornem informações úteis que possam apoiar a tomada de decisão, torna-se necessário um considerável esforço para integrá-los.

É neste ambiente que a tecnologia Data Warehouse (para português literal: Armazém de Dados) vem ganhando destaque ao passo que oferece mecanismos eficientes para integrar dados internos e externos de maneira flexível em uma estrutura única, facilitando assim a transformação de dados em informações que apoiarão processos decisórios.

Com um grande volume de dados históricos integrados, sistemas OLAP (On-line Analytical Processing) e técnicas de data mining oferecem mecanismos práticos e sofisticados para análise de dados e descoberta de conhecimento.

O Data Warehouse, assim como qualquer outra tecnologia da informação, tem como objetivo suportar o negócio e portanto, deve estar alinhado com as estratégias e objetivos gerais da organização. É uma tecnologia que envolve altos recursos e investimentos financeiros, mas vem provando ser uma ferramenta muito poderosa para transformar dados em novas oportunidades de negócios dentro das organizações.

1. A EVOLUÇÃO DOS SISTEMAS DE APOIO A DECISÃO

1.1. Um breve histórico

As origens do processamento dos sistemas de apoio a decisão remontam aos primórdios dos computadores, e a sua evolução pode ser estruturada em alguns estágios entre 1960 e os dias de hoje, conforme veremos a seguir.

No início da década de 1960 o mundo da computação consistia na criação de aplicações individuais que eram executadas sobre arquivos mestres, caracterizadas por programas e relatórios. Nesta primeira fase, o uso de cartões perfurados era comum, e as fitas magnéticas eram os dispositivos mais adequados para o armazenamento de grandes volumes de dados a baixo custo. Ainda assim, as fitas magnéticas, apresentavam o inconveniente de terem que ser acessadas sequencialmente.

Por volta de 1965, com crescimento exponencial dos arquivos mestres e das fitas magnéticas, surgiram enormes quantidades de dados redundantes dentro das empresas. Como conseqüências da massiva redundância nascem alguns problemas críticos como: a complexidade de manutenção e desenvolvimento de novos programas; a quantidade de hardware necessária para manter todos os arquivos mestres e a necessidade de sincronizarem dados a serem atualizados.

Os diversos problemas com os arquivos mestres já se tornavam sufocantes, e já era mais do que hora de uma nova tecnologia de armazenamento e acesso de dados entrarem em cena, e assim aconteceu. No início da década de 1970, surge a tecnologia DASD (Direct Access Storage Device), substituindo as fitas magnéticas pelo armazenamento em disco. A principal vantagem que esta tecnologia trazia, como o próprio nome já diz, era o mecanismo de acesso direto ao endereço do registro desejado.

Com o DASD surgiu um novo tipo de software conhecido como SGBD (Sistema de Gerenciamento de Banco de Dados), que tinha o objetivo de tornar o armazenamento e o acesso a dados no DASD mais fáceis para o programador. As vantagens tecnológicas da tecnologia DASD somadas com a praticidade do SGBD chegam para solucionar o problema dos arquivos mestres. Ainda neste mesmo contexto é que surge o conceito de banco de dados como: uma única fonte de dados para todo o processamento.

Aproximadamente em 1975 surgiu o processamento de transações on-line sobre bancos de dados, abrindo perspectivas totalmente novas. Agora o computador poderia ser usado para tarefas que antes não eram viáveis como controlar sistemas de reservas, sistemas de caixas bancários, sistemas de controle de produção e outros.

Até o início da década de 1980, novas tecnologias, como os PCs (Personal Computer) e as L4Gs (Linguagens de Quarta Geração) começaram a emergir, e o usuário final passou a controlar diretamente os sistemas e os dados fora do domínio do clássico processamento de dados. Em meio a este cenário, surge a noção de que é possível utilizar os dados para outros objetivos além de atender ao processamento de transações on-line de alta performance, aliando assim, mais do que nunca os técnicos e as pessoas de negócio.

Atualmente conhecidos como SADs, os MIS (Management Information System ou sistemas de informações gerenciais) tornaram-se viáveis, e a tecnologia e os dados até então utilizados exclusivamente para direcionar decisões operacionais detalhadas, passaram também a apoiar e direcionar decisões gerenciais. Surge então um paradigma: um único banco de dados que poderia atender simultaneamente ao processamento de transações e ao processamento analítico, ou de SAD (INMON, 1997).

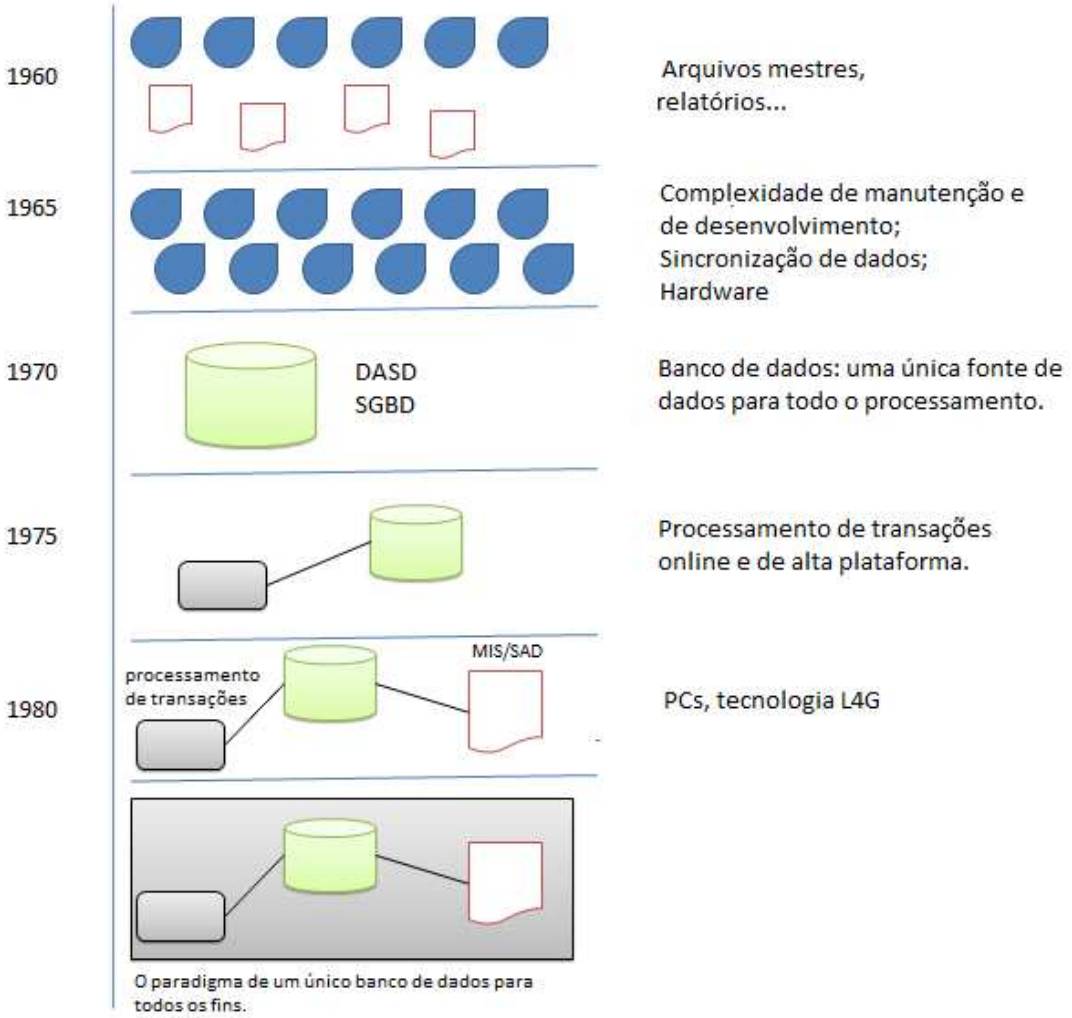

Figura1 – Os primeiros estágios evolucionários do ambiente projetado.

Posterior ao advento das transações online de alta performance foi a vez dos programas de extração se difundir pelo ambiente de processamento de informações. Estes são programas razoavelmente simples, que varrem um banco de dados, usando alguns critérios de seleção, e, ao encontrar dados que atendam estes critérios, transporta-os para outro arquivo ou banco de dados.

Com o programa de extração, era possível retirar os dados do ambiente de processamento online, isentando-o de qualquer ônus em termos de performance quando os dados precisavam ser analisados. Além disso, os dados extraídos passavam a ser de controle do usuário final, adquirindo assim, maior autonomia sobre eles e garantindo a integridade no ambiente de processamento de transações online. Por estas e outras vantagens, os programas de extração podiam ser encontrados dentro de grande parte das empresas a partir da segunda metade da década de 1980.

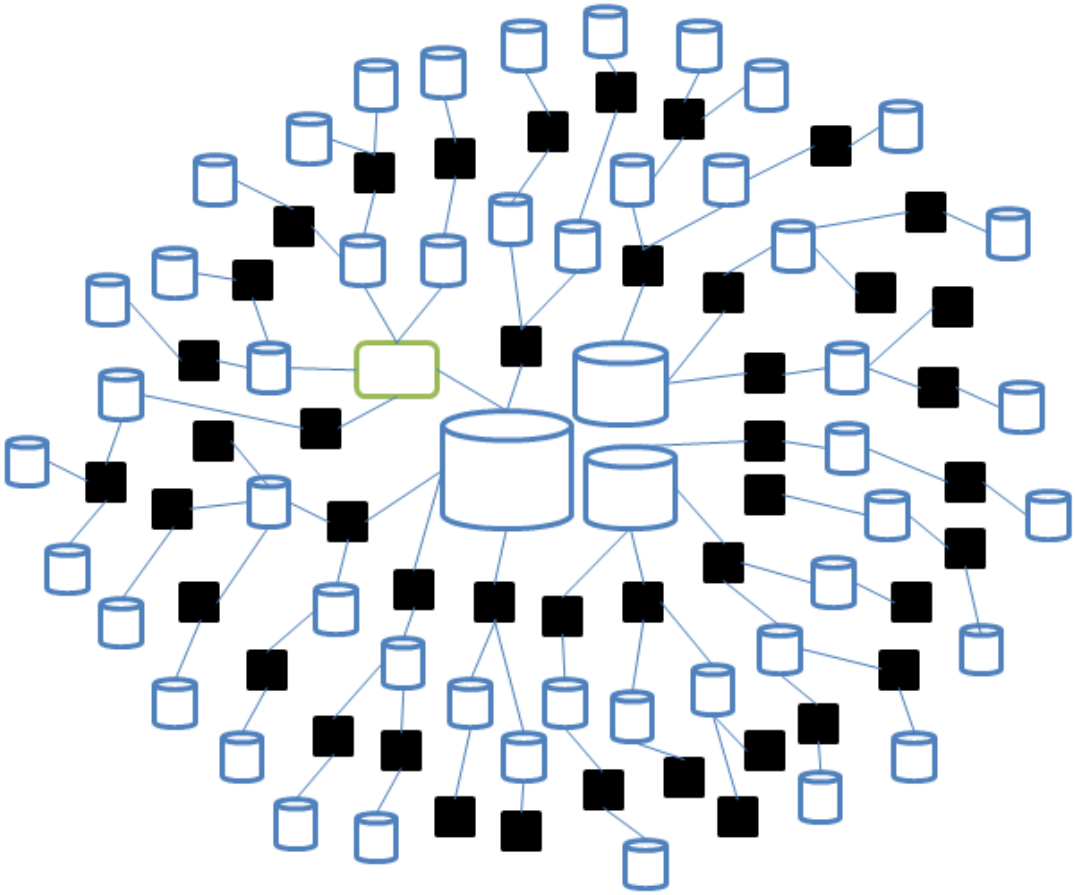

Figura 2 – Arquitetura de desenvolvimento espontâneo.

Contudo, os programas de extração abriram margens para a arquitetura de desenvolvimento espontâneo (conhecida como “teia de aranha”), que consiste no processamento de extração fora de controle. Desta forma, primeiro havia as extrações. Em um outro nível, extrações das extrações, e assim por diante. Até o início da década de 1990, era comum encontrar empresas que adotavam esta postura. Quanto maior e mais madura a organização, piores se tornam os problemas relacionados com a arquitetura de desenvolvimento espontâneo. Dentre os seus principais problemas, podemos citar:

- **Falta de credibilidade dos dados e informações**: Uma crise de credibilidade pode ocorrer, por exemplo, quando relatórios de dois departamentos diferentes, geram informações conflitantes e sem integridade. Alguns fatores que abrem margens para a falta de credibilidade são a ausência de parâmetros de tempo de dados, diferencial algorítmico dos dados, diferentes níveis de extração, diferentes

fontes de dados externos e falta de uma fonte de dados única e comum com a qual começar.

- **Problemas de produtividade:** A produtividade é extremamente onerosa ao passo que exige um trabalho exaustivo e maçante na hora de localizar, analisar e reunir os dados necessários para produzir um relatório corporativo, pois na maioria das vezes, tais dados estão disponíveis em arquivos dispersos e com layouts distintos. Após a localização e análise dos dados que serão utilizados, a produtividade é afetada também na etapa de compilação destes dados para o relatório, pois existe a necessidade de criação de muitos programas, cada um customizado de acordo com a sua fonte.
- **Impossibilidade de transformar dados em informações:** Uma das causas de tal impossibilidade se resume à falta de integração entre os sistemas de processamento transacionais que provem os dados necessários para cumprir o atendimento de uma dada solicitação. Assim, tentar extrair informações dessas aplicações segundo um critério geral é quase impossível. Um segundo obstáculo importante consiste em que não há dados históricos suficientes armazenados nas aplicações para satisfazer às necessidades da solicitação do SAD.

Portanto, o status quo da arquitetura de desenvolvimento espontâneo, no qual se encontra grande parte das organizações atualmente, simplesmente não basta para atender as necessidades do futuro, trazendo assim, a necessidade de uma mudança de enfoque para um possível ambiente projetado (INMON, 1997).

No cerne do ambiente projetado está a percepção de que há fundamentalmente duas espécies de dados, conforme podemos observar no quadro abaixo:

DADOS PRIMITIVOS/OPERACIONAIS	DADOS DERIVADOS/DADOS SAD
• baseados em aplicações • detalhados • exatos em relação ao tempo de acesso • atendem à comunidade funcional • podem ser atualizados • são processados repetitivamente • requisitos de processamento conhecidos com antecedência • a performance é fundamental	• baseados em assuntos ou negócios • resumidos, ou refinados • representam valores de momentos já decorridos ou instantâneos • atendem à comunidade gerencial • não são atualizados • processados de forma heurística • requisitos de processamento não são conhecidos com antecedência • a performance é atenuada

• voltados para transações • o controle de atualizações é atribuição de quem tem a posse • alta disponibilidade • gerenciados na sua totalidade • não contemplam a redundância • estrutura fixa; conteúdos variáveis • pequena quantidade de dados usada em um processo • atendem às necessidades cotidianas • alta probabilidade de acesso	• acessados um conjunto por vez • o controle de atualizações não é problema • disponibilidade atenuada • gerenciados por subconjuntos • redundância não pode ser ignorada • estrutura flexível • grande quantidade de dados usada em um processo • atendem às necessidades gerenciais • baixa, ou modesta probabilidade de acesso

Tabela 1 – No ambiente projetado, toda noção de dados é alterada.

Como se pode notar há uma grande diferença entre os dados primitivos e derivados. Levando em consideração todas essas diferenças, é realmente espantoso acreditar que a comunidade de processamento de informações tenha pensado que dados primitivos e dados derivados pudessem compartilhar um mesmo ambiente em um banco de dados único.

1.2. O ambiente projetado

Existem quatro níveis na arquitetura do ambiente projetado: o operacional, o atômico ou Data Warehouse (DW), o departamental e o individual.

O nível operacional de dados contém apenas dados primitivos e atende à comunidade de processamento de transações de alta performance. O Data Warehouse contém dados primitivos que não são atualizados e dados derivados. O nível departamental de dados praticamente só contém dados derivados. E o nível individual de dados é onde a maior parte das análises heurísticas é feito (INMON, 1997). A Figura 3 mostra os tipos de consulta para os quais os diferentes níveis de dados podem ser usados.

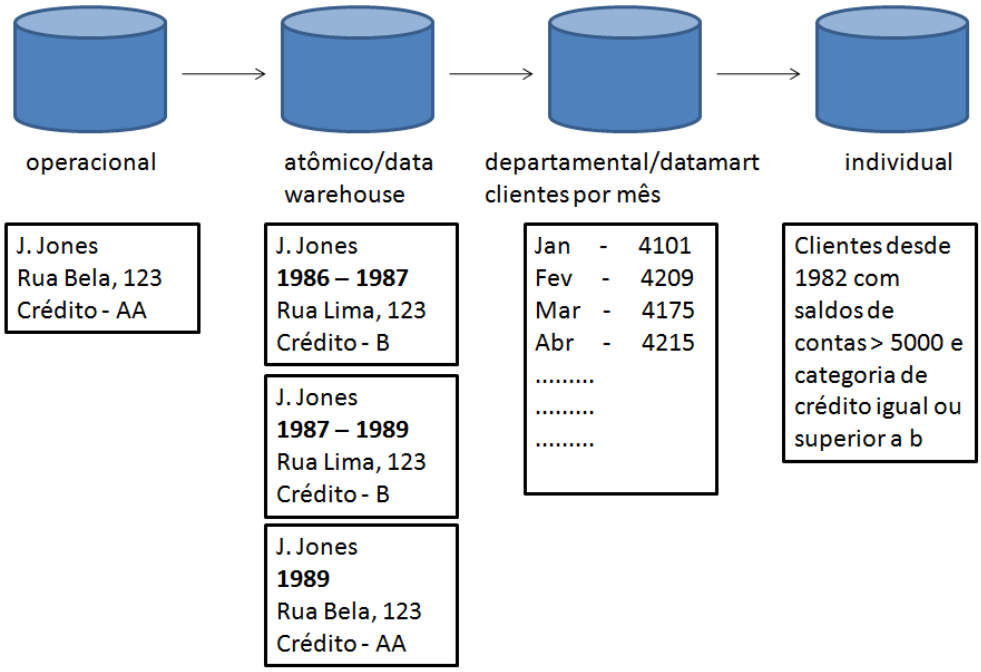

Figura 3 – Exemplo: tipos de consultas sobre um cliente de diferentes níveis de dados.

Examinando o exemplo da figura acima, podemos observar como os dados estão disponíveis ao longo da arquitetura. No nível operacional, o registro terá dados de valor atual, sendo que, se as informações referentes a J. Jones sofrerem alterações, o registro existente neste nível, será alterado para refletir os novos dados corretos. No ambiente de data warehouse, há vários registros referentes a J. Jones que não se sobrepõem, sendo que cada um desses possui um elemento de tempo associado. No ambiente departamental, contém informações úteis aos diversos departamentos de uma empresa. E por fim, o nível individual possui dados geralmente temporários e de pequenas proporções, onde são realizadas as análises heurísticas. Normalmente os níveis individuais de dados são encontrados em PCs.

Apenas para complementar os fundamentos do ambiente projetado, um fator que não fica explicito na figura 3, é justamente a idéia de integração existente neste ambiente. À medida que os dados passam do ambiente operacional para o ambiente de data warehouse, eles vão sendo integrados, permitindo desta forma, uma visão corporativa sobre dados e facilitando a geração de informação.

2. CONCEITOS E DEFINIÇÕES

2.1. Conceito e Características de Data Warehouse

Conforme acompanhamos no capítulo anterior, com a evolução tecnológica e a clara distinção entre dados primitivos e dados derivados, surge o ambiente projetado**.** No coração do ambiente projetado encontra-se o Data Warehouse, e este é o alicerce do processamento dos SADs (INMON, 1997).

Com a utilização dos sistemas transacionais de alta performance, as grandes empresas passaram a deter um enorme volume de dados provenientes de inúmeros sistemas espalhados por ela. A falta de integração entre os diversos sistemas e de uma estrutura adequada que permita cargas históricas, torna extremamente exaustiva a busca por informações que permita a tomada de decisões embasada num histórico dos dados.

Neste contexto, surge o conceito de Data Warehouse como uma solução utilizada para identificação de tendências, de modo a posicionar a empresa estrategicamente para ser mais competitiva e consequentemente maximizar os lucros diminuindo o índice de erros na tomada de decisão. Isso se torna possível através de uma estrutura capaz de integrar e manter um histórico de dados dispostos dimensionalmente, fazendo com que o mesmo dado ou informação possa ser visualizado por várias dimensões diferentes.

Segundo Inmon (1997), um Data Warehouse é um conjunto de dados baseado em assuntos, integrado e não-volátil, e variável em relação ao tempo, de apoio a decisões gerenciais.

Para cada uma das características definidas por Inmon (1997), pode-se entender:

- **Coleção de dados** como a o volume de dados provenientes dos sistemas transacionais (ambiente operacional), que serão utilizados para a obtenção de novas informações que auxiliem a tomada de decisão.
- **Baseado em assuntos** como uma estrutura assumida pelo DW com intuito de organizar os dados disponíveis em torno de suas diversas aplicações em assuntos ou negócios relacionados à empresa.
- **Integrados** como conseqüência da fase de extração, transformação e carga de dados no DW que será explicada posteriormente. Quando os dados passam do

ambiente operacional baseado em aplicações para o DW, todas as inconsistências são desfeitas, mantendo padronização e integridade entre os dados.

- **Não voláteis, pois** diferente dos dados operacionais, que são acessados e tratados um registro por vez e sofrem atualizações, os dados do DW são carregados em grandes quantidades, somente acessados e não sofrem atualizações.
- **Variáveis com o tempo**, que pode se manifestar de diversas formas. Primeiramente, o horizonte de tempo válido para o data warehouse é significantemente maior do que o dos sistemas operacionais. Enquanto um horizonte de tempo de 60 a 90 dias é normal para os sistemas operacionais, um tempo de horizonte de 5 a 10 anos, pode ser consideravelmente razoável para o Data Warehouse. Além desse primeiro aspecto, os bancos de dados transacionais contêm dados de valor corrente, cuja exatidão é valida para o momento de acesso, e que podem ser atualizados. Dados presente no DW podem ser considerados como uma série sofisticada de instantâneos, capturados num determinado momento. Por fim, a estrutura de chave dos dados operacionais não necessariamente é composta por elementos de tempo (ano, mês, dia, etc.), enquanto no Data Warehouse, sempre encontramos algum elemento de tempo compondo a chave.

Com a chegada do DW, novos métodos de estruturação de dados e novas tecnologias, tanto para armazenamento, como para recuperação de informações, passaram a ser necessários. Graças aos avanços nos bancos de dados relacionais, no processamento paralelo e na tecnologia distribuída, a tecnologia da informação pode permitir que as organizações implantem um projeto de Data Warehouse, desde que seja considerada a relação custo/benefício.

2.2. Diferenças entre o processamento transacional e analítico

O ambiente de processamento de dados analíticos, ou o Data Warehouse difere bastante do ambiente de dados transacionais ou operacionais, baseado em OLTP (On-Line Transaction Processing). Em resumo, podemos assumir que os sistemas OLTP servem como fonte de dados para o ambiente de Data Warehouse, enquanto o processamento analítico OLAP (On-Line Analytical Processing) ajuda a analisá-los.

Complementando a idéia exibida na figura 4, o OLTP é caracterizado por um grande número de transações on-line curtas (por exemplo, INSERT, UPDATE, DELETE), onde os fatores de missão crítica são basicamente a performance no tempo de resposta e a integridade de dados irredundantes em ambientes multi-acesso. Conforme já dito anteriormente, como os bancos de dados operacionais atendem apenas os processos do negócio e são utilizados em operações diárias da empresa, as informações históricas não ficam armazenadas por muito tempo.

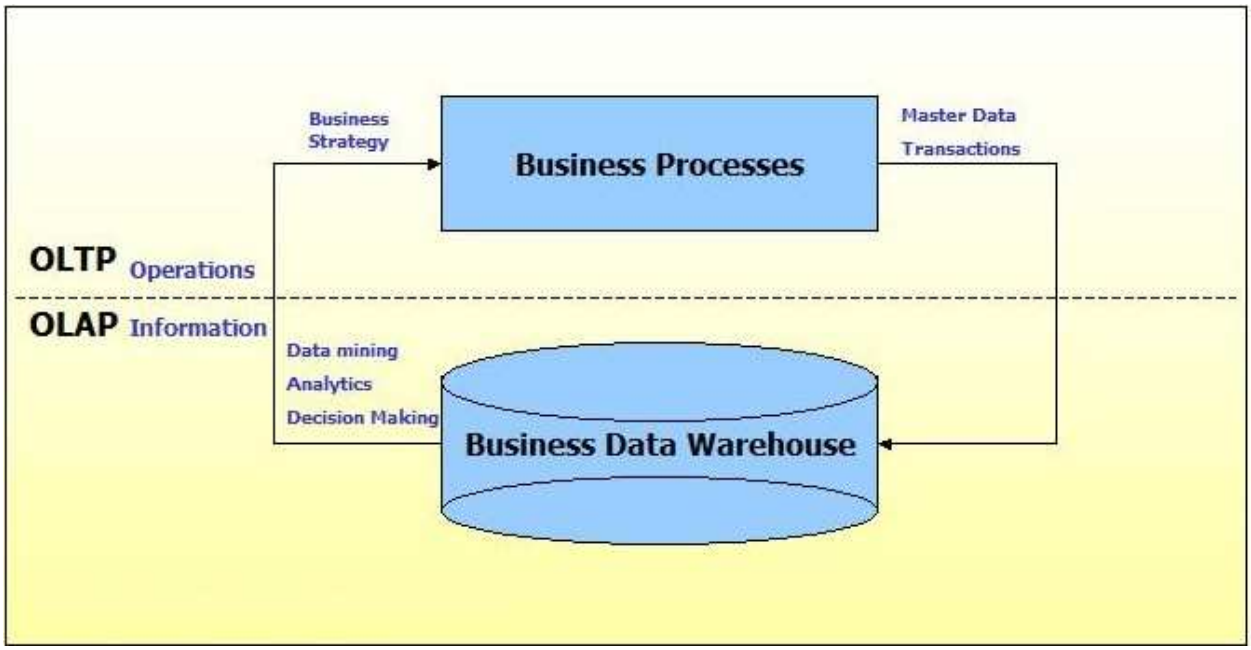

Figura 4 – OLAP vs. OLTP

Ao contrário do OLTP, o ambiente de processamento analítico é caracterizado por consultas complexas, estruturadas e frequentes, envolvendo agregação ou relacionamento de dados para gerar informações que apoiarão processos decisórios. No OLAP, o tempo de resposta não é um fator de missão crítica, apesar de ser determinante para a produtividade no ambiente de SAD. Como o próprio nome já diz, este tipo de processamento é amplamente utilizado para análises sobre o negócio e para a aplicação de técnicas de Data Mining.

Para consolidar a compreensão dos conceitos apresentados acima, a tabela seguinte resume as principais diferenças entre o ambiente de banco de dados transacionais e o ambiente analítico ou Data Warehouse:

Características	Ambiente operacional	Data Warehouse
Objetivo	Atender e controlar operações diárias do negócio.	Analisar o negócio, gerar novas oportunidades de negócio.
Processamento	OLTP	OLAP
Uso	Operacional	Informativo, Analítico
Fonte de dados	Dados operacionais.	Dados consolidados, provenientes de muitos bancos de dados transacionais
Unidade de trabalho	Inclusão, alteração, exclusão.	Carga e consultas complexas
Visão dos dados	Momento instantâneo dos processos de negócio em andamento.	Visão multidimensional dos vários tipos de atividades do negócio.
Velocidade/Performance	Muito rápido. Performance é fator de missão crítica.	Depende do volume de dados envolvidos. Apesar de não ser fator de missão crítica, performance de consultas pode ser melhor com a criação de índices.
Número de usuários	Milhares	Centenas
Tipo de usuário	Operadores	Comunidade gerencial
Interação do usuário	Somente pré-definida	Pré-definida e ad-hoc
Volume	Megabytes – gigabytes	Gigabytes – terabytes
Histórico	60 a 90 dias	5 a 10 anos
Redundância	Não ocorre	Ocorre
Atualização	Contínua (tempo real)	Periódica (em batch)
Integridade	Transação	A cada atualização
Intenção dos índices	Localizar um registro	Aperfeiçoar consultas

Tabela 2 – Diferenças entre o ambiente operacional e o Data Warehouse.

2.3. Granularidade

A mais importante questão de projeto que o desenvolvedor do data warehouse precisa enfrentar refere-se à definição da granularidade do data warehouse, ou seja, o nível de detalhe ou de resumo dos dados existentes no data warehouse. Quando a granularidade de um data warehouse é apropriadamente estabelecida, os demais aspectos de projeto e implementação fluem tranqüilamente; quando ela não é estabelecida, todos os outros aspectos se complicam (INMON, 1997).

A granularidade diz respeito ao nível de detalhe ou de resumo contido nas unidades de dados existentes no data warehouse. Quanto mais detalhe, menor o nível de granularidade. Quanto menos detalhe, maior o nível de granularidade (INMON, 1997).

O principal motivo por que a granularidade é uma questão crucial para o projeto do data warehouse, consiste no fato de que afeta tanto no volume de dados do data warehouse quanto nas consultas que podem ser atendidas (INMON, 1997).

É evidente que a questão de espaço é um problema em um data warehouse, e um alto nível de granularidade consiste em uma maneira muito mais eficiente de representação dos dados, ao passo que ocorre uma economia em termos de DASD, de número de índices necessários para manter a performance e de recursos de processamento para tratamento dos dados (INMON, 1997).

Por outro lado, à medida que o nível de granularidade se eleva, há uma correspondente diminuição da possibilidade de utilização dos dados para atender a consultas. Em outras palavras, com um alto nível de granularidade, o número de questões a que os dados podem satisfazer é limitado. Portanto, o volume de dados contidos no data warehouse deve ser balanceado com o nível de detalhe das consultas que ele pretende atender, conforme ilustrado na figura abaixo (INMON, 1997).

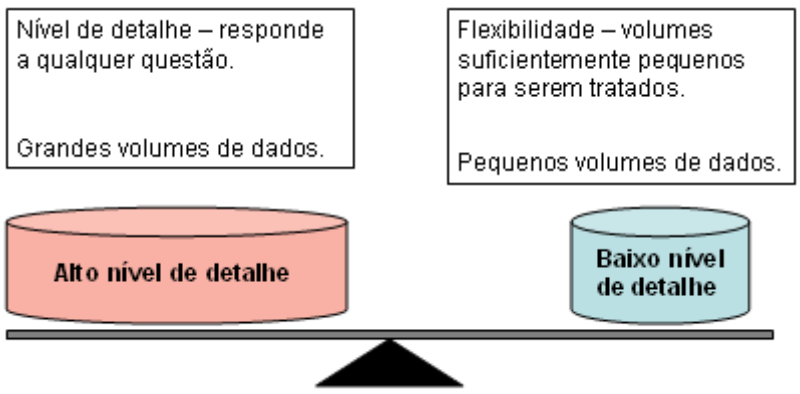

Figura 5 – O balanceamento da granularidade de dados no data warehouse.

Durante o processamento de SAD, como é comum no ambiente de data warehouse, dificilmente um evento isolado é examinado. É mais comum ocorrer a utilização de uma visão de conjunto de dados, onde um grande volume de dados são analisados. Este tipo de consulta pode ser respondida tanto pelo nível alto

quanto pelo nível baixo de granularidade, com uma tremenda diferença de utilização de recursos. Sendo assim, o uso dos dados do nível alto de granularidade é muito mais eficiente se ele apresenta o detalhamento suficiente (INMON, 1997).

Existe ainda a possibilidade de utilizar um nível duplo de granularidade (níveis duais de granularidade). Esta técnica se enquadra nos requisitos da maioria das empresas. São criadas duas camadas: uma camada para os dados levemente resumidos e outra para os dados históricos. Com a criação de dois níveis de granularidade, é possível atender a todos os tipos de consultas, com um melhor desempenho, visto que a maior parte do processamento analítico dirige-se aos dados levemente resumidos que são compactos e de fácil acesso e para as ocasiões em que um maior nível de detalhe deve ser analisado, existe o nível de dados históricos, o qual é complexo e de alto custo (INMON, 1997).

2.4. Metadados

Toda e qualquer informação no ambiente DW que não são os dados propriamente ditos, são chamados metadados. Estes são como uma enciclopédia para o DW. Eles estão presentes em uma variedade de formas e formatos para suportar as necessidades desiguais dos grupos de usuários técnicos, administrativos e de negócio do DW (KIMBALL, 2002).

Metadados nada mais são além de dados sobre dados, e fazem parte do meio de processamento de informações há tanto tempo quanto os programas e os dados. Portanto, no mundo dos data warehouses é que os metadados assumem um novo nível de importância ao passo que é por meio deles que a utilização mais produtiva do data warehouse pode ser alcançada (INMON, 1997).

Os usuários de DW precisam conhecer a estrutura e o significado dos dados do DW para poder examinar os dados, o que não ocorre em sistemas, onde os usuários interagem com as telas do sistema sem precisar conhecer como os dados são mantidos pelo banco de dados.

Outra razão para a importância dos metadados é concernente ao gerenciamento do mapeamento entre o ambiente operacional e o ambiente de data warehouse. À medida que os dados passam do ambiente operacional para o ambiente de data warehouse eles são submetidos a significativas transformações

através de filtros, conversões, resumos e alterações estruturais. Essas transformações precisam manter um rigoroso acompanhamento, e os metadados do DW constituem um local ideal para isso (INMON, 1997).

Mais uma tarefa dos metadados no ambiente de data warehouse é a de manter o acompanhamento das alterações das estruturas de dados ao longo do tempo.

Segundo Inmon (1997), os metadados englobam o DW e mantém informações sobre “o que está aonde” no DW. Tipicamente os aspectos 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.

2.5. Data Mart

Segundo Inmon (1997), um Data Mart é uma coleção de dados relacionados a alguma área da empresa, organizados para dar suporte à decisão e baseados nas necessidades de um departamento. Em outras palavras, um Data Mart representa um subconjunto de dados de um Data Warehouse que possibilita um visão mais especializada, limitada e focada em necessidades de unidades específicas de negócio ao invés da corporação inteira.

Muitas empresas ingressam em projetos de Data Mart focando em atender necessidades especiais de pequenos grupos dentro da organização. Os Data Marts são bem aceitos entre as organizações, pois exigem menor tempo para implementação e investimento em infra-estrutura, trazem resultados mais rapidamente e são escaláveis até um DW.

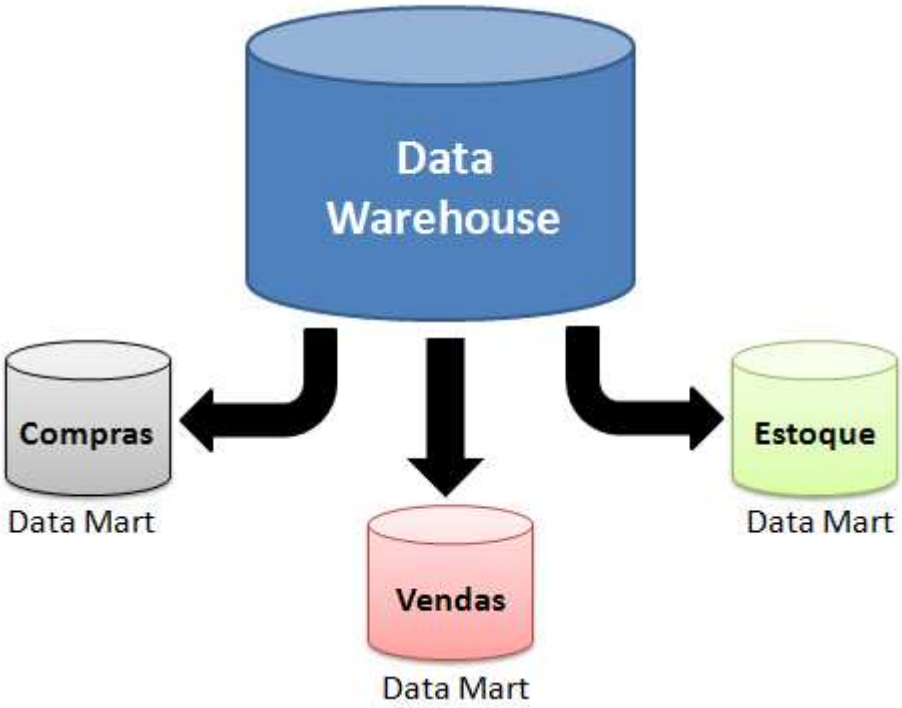

Figura 6 – Data Marts por unidades específicas de negócio.

Segundo Ralph Kimball (2002), os Data Marts não devem ser departamentais, mas sim orientados aos dados ou fontes de dados. Kimball (2002) defende também que as empresas devem construir vários Data Marts para posteriormente integrá-los, compondo assim o DW. Os Data Marts devem ser orientados por assunto, e pontos de conexão entre eles, as Tabelas Fato.

De acordo com Inmon (1997), as empresas devem construir primeiro o Data Warehouse, modelando toda a empresa até chegar a um modelo corporativo, e em seguida construir os Data Marts por assunto ou departamento. Os Data Marts devem atender os diversos departamentos de uma empresa, gerando dados íntegros e corporativos. De acordo com Inmon, o processo inverso priorizaria as necessidades de cada departamento isolado às necessidades da empresa como um todo, além de poder gerar redundância nos dados em diversos sistemas, o consumo exagerado de recursos de produção e a falta de integração dos dados dispostos em Data Marts diferentes.

A construção do Data Mart pode seguir duas abordagens distintas:

- Top-Down ocorre 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. Conforme podemos ver na figura 7:

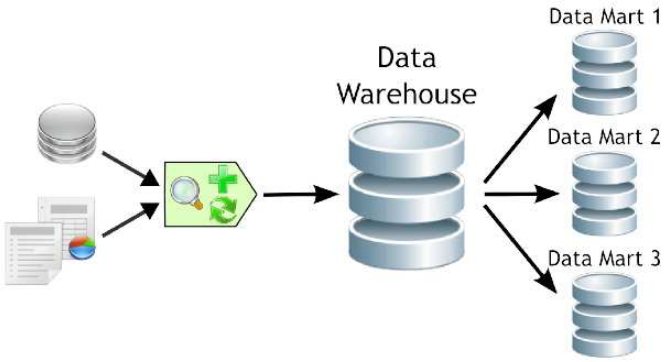

Figura 7 – Abordagem Top-Down

- Botton-up: 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 em um Data Warehouse.

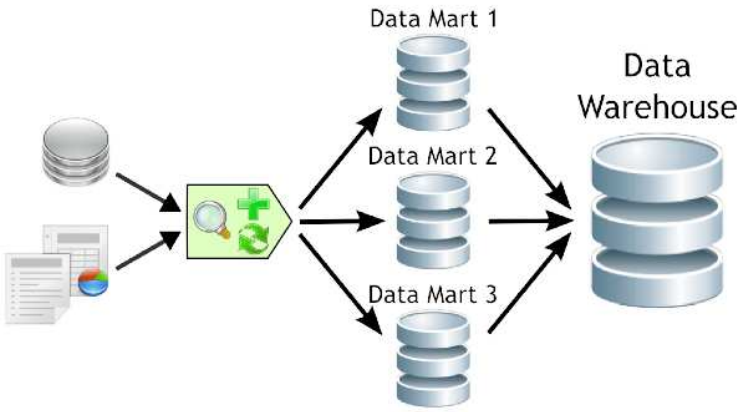

Figura 8 – Abordagem Botton-up

3. MODELAGEM DIMENSIONAL

3.1. Dimensionalidade

Nos bancos de dados relacionais, onde a performance em tempo de resposta é um fator crítico de sucesso, a redundância de dados é eliminada pelo processo de normalização, tornando assim as transações e consultas mais simples e rápidas, e oferecendo a melhor alternativa para atender as tarefas do dia-a-dia.

Ao contrário do ambiente operacional, no DW são processadas consultas menos frequentes, mais complexas e envolvendo um grande volume de dados. Por conta disso, não se torna viável a aplicação de normalização das tabelas, uma vez que no DW ocorrem poucas transações concorrentes e cada transação acessa um grande número de registros.

Portanto, enquanto o banco de dados relacional utiliza a modelagem de Entidade-Relacionamento, o DW utiliza o que chamamos de modelagem dimensional. A modelagem dimensional é uma técnica que aplicamos para ter uma visão multidimensional dos dados.

Segundo Kimball (2002), a modelagem dimensional é uma disciplina de design que transpassa a modelagem relacional e a realidade de dados de texto e números. Um modelo dimensional é composto basicamente pela tabela de fatos e pelas tabelas de dimensões. A tabela fato traz o resultado da consulta e valores de medição. As restrições, objeções e questionamentos ficam nas tabelas dimensões, que trazem informações textuais sobre o valor medido na tabela fato.

3.2. Tabela de Fatos

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

A tabela de fatos é sempre esparsa, ou seja, possui um número relativamente pequeno de todas as combinações possíveis de valores de chaves. Por exemplo, no caso de um banco de dados de uma companhia aérea, a presença de todas as combinações possíveis representaria que todos os clientes voam todos os dias, e em todos os vôos feitos pela companhia, o que na prática é impossível. Por isso podemos dizer que esse banco é extremamente esparso, pois uma porcentagem muito pequena de todas as combinações possíveis de clientes, número do vôo e dia aparecerão nele (HOKAMA, 2004).

Um ponto a ser considerado, é que a tabela de fatos deve representar uma unidade do processo de negócio, que não mistura diferentes assuntos numa mesma tabela de fatos.

Os atributos encontrados em uma tabela de fatos podem ser classificados em:

- Métricas aditivas: são numéricas e permitem operações como soma, subtração e média de valores por todas as dimensões existentes, por exemplo: quantidade ou valor total de produtos vendidos considerando data, produto e loja.
- Métricas semi-aditivas: não podem ser somadas com relação a todas as dimensões, sendo que faz realizar a soma em apenas uma dimensão.
- Métricas não-aditivas: não faz sentido realizar operações com os valores em nenhuma dimensão.

Um exemplo de uma tabela de fatos pode ser visualizado na tabela de vendas de uma empresa ilustrada na figura 9:

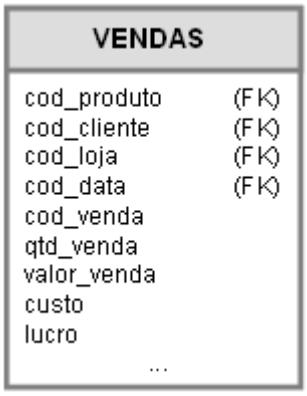

Figura 9 – Tabela de fato

3.3. Tabela de Dimensão

A tabela de dimensão contém as descrições textuais do negócio, e possui as informações necessárias para análises ao longo de dimensões. Seus atributos são fonte das restrições das consultas, agrupamento dos resultados, e cabeçalhos para relatórios. As dimensões são os aspectos pelos quais se pretende observar as métricas relativas ao processo que está sendo modelado (KIMBALL, 2002).

A qualidade do banco de dados é proporcional à qualidade dos atributos de dimensões, portanto deve ser dedicados tempo e atenção a sua descrição, ao seu preenchimento e a garantia da qualidade dos valores em uma coluna de atributos (KIMBALL, 2002).

É muito importante que os atributos das tabelas de dimensão sejam preenchidos com valores descritivos ao invés de códigos sem sentido, criptografados ou abreviados (KIMBALL, 2002).

A tabela de dimensão geralmente é bem menor que a tabela de fatos. Cada dimensão é definida com uma chave primaria única, que representa a integridade no relacionamento com a tabela de fatos.

A figura 10 mostra exemplo da tabela de dimensão.

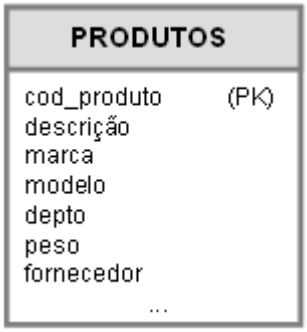

Figura 10 – Tabela de dimensão

3.4. Tabelas Agregadas

Um fator que reflete diretamente na produtividade das aplicações de análise de dados, como já vimos anteriormente, é o tempo de resposta ao usuário. Muitas vezes esse tempo é considerado crítico devido ao grande volume de dados envolvido nas consultas. Uma técnica que pode oferecer ganhos significativos de

performance é a criação de tabelas agregadas, que consiste em criar tabelas com os dados das tabelas de fatos, porém considerando maior granularidade para estes.

Desta forma, as tabelas agregadas evidentemente serão menores e mais sumarizadas, facilitando o acesso aos dados e agilizando o processo de tomada de decisão. Por outro lado, o espaço alocado para o armazenamento dos dados em um estado pré-processado será maior.

É muito importante analisar o ambiente antes de definir quais agregações serão criadas. É necessário realizar estatísticas de consultas para considerar quais os requisitos que são mais frequentes e quais consultas são mais críticas para posteriormente definir quais agregadas serão mais úteis.

Cada nova carga de dados no data warehouse acarretará no recálculo de toda ou pelo menos parte das tabelas agregadas, para que ela contemple os novos dados incluídos.

Existem duas formas de atualizar uma tabela agregada:

- Agregação completa: consiste em recriar a tabela agregada toda vez que houver uma carga na tabela de fatos;
- Agregação incremental: os dados existentes na tabela são alterados e apenas os dados novos são inseridos.

A utilização dessas tabelas deve ser sempre monitorada para verificar se realmente está se obtendo vantagem com a sua existência. O benefício gerado pela criação de uma agregada é calculado baseado na redução do volume de dados e na freqüência de sua utilização. Deve-se sempre levar em conta a possibilidade de uma agregada ser extinta e a manutenção que isso causaria a todo o processo (HOKAMA, 2004).

3.5. Técnicas de Modelagem

Existem técnicas específicas para modelagem dimensional, sendo as mais utilizadas são a Star Schema e a Snow Flake, as quais serão mais bem explanadas nos itens a seguir.

3.5.1. Star Schema

O esquema estrela possui uma estrutura razoavelmente simples com poucas tabelas e relacionamentos bem definidos, se aproximando bastante do modelo de negócio, o que facilita bastante na leitura e compreensão até mesmo de usuários finais que não estão adaptados com estruturas de bancos de dados. Desta forma, o esquema estrela facilita a criação de consultas complexas, facilitando que estas sejam realizadas de forma intuitiva pelo próprio usuário.

A palavra “estrela” está associada à forma como as tabelas ficam dispostas no modelo. Como podemos notar na figura 11, o esquema estrela consiste em uma tabela central, a tabela de fatos, que se relaciona com diversas outras tabelas, que são as tabelas de dimensão.

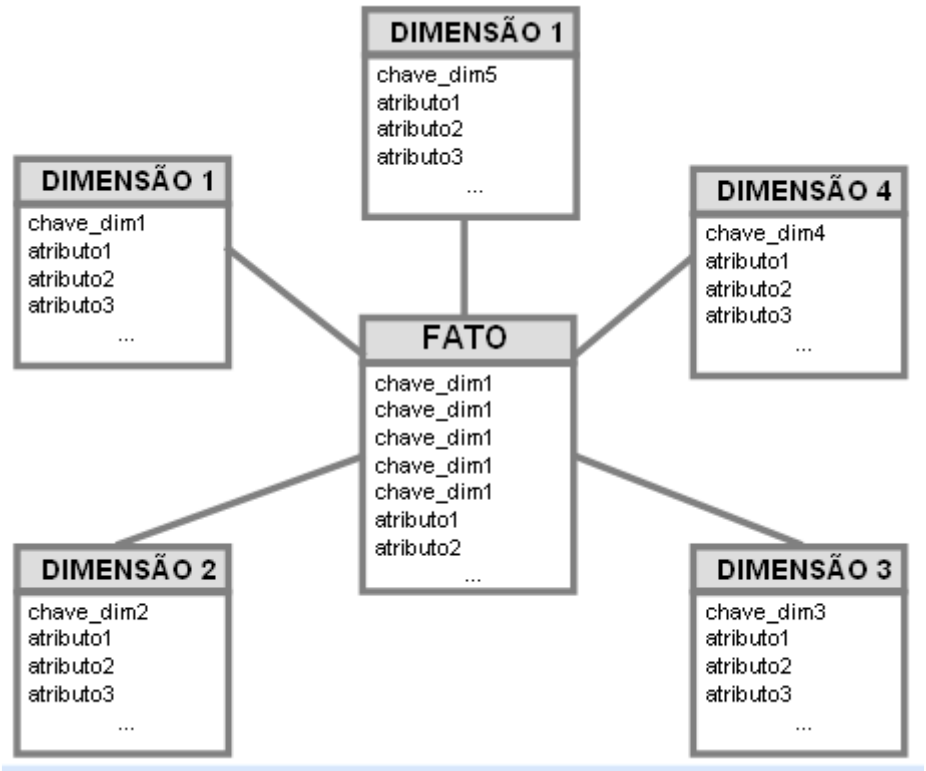

Figura 11 – Modelo Star Schema

3.5.2. Snow Flake

O modelo de flocos de neve nada mais é do que uma variação do esquema estrela com suas tabelas de dimensão normalizadas até a terceira forma normal. Sendo assim, são retirados das tabelas de dimensões os campos que são funcionalmente dependentes de outros campos que não são chaves.

A aplicação de normalização nas tabelas de dimensão pode aumentar a complexidade do modelo, diminuindo assim a compreensão do mesmo por parte do usuário. Além disso, a performance das consultas tende a diminuir devido ao maior número de junções.

De acordo com Ralph Kimball (2002), os projetistas "bem-intencionados" devem resistir à tentação de transformar esquemas estrela em esquemas 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.

3.6. Cubo de dados

Segundo Kimball (2002), uma idéia fundamental da modelagem dimensional é que quase todos os tipos de dados de negócio podem ser representados por um tipo de cubo de dados, onde as células deste cubo contêm valores medidos e os lados do cubo definem as dimensões dos dados. Pode-se ter mais que três dimensões, tecnicamente chamado de hipercubo, apesar de normalmente os termos cubo e cubo de dados serem usados como sinônimos de hipercubo.

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

Em outras palavras, um cubo é uma estrutura multidimensional que consolidam e facilitam a análise de dados. Os cubos de dados são formados por

uma tabela de fatos e definidos por várias tabelas de dimensões, conforme podemos observar na figura 12. Considerando o cenário de uma instituição bancária, por exemplo, poderíamos montar um cubo contendo informações de aquisição de crédito, considerando as dimensões de região, produto, período e medidas. As medidas seriam os valores que iriam compor as células do cubo, como por exemplo, quantidade de contas abertas, valor de limite de crédito ou até mesmo valor de perda por inadimplência.

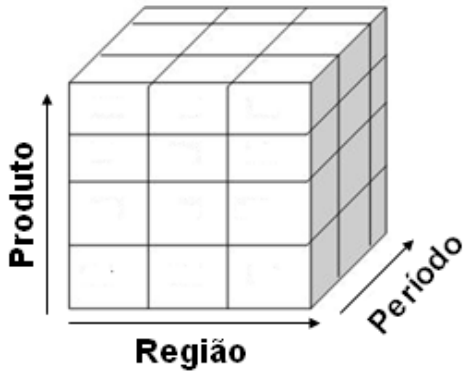

Figura 12 – Exemplo: cubo de dados considerando dimensões de produto, região e período.

Para se visualizar a análise multidimensional em cubo utiliza-se a técnica de slice-dice, ou seja, fatiar e cortar o cubo separando partes de um cubo (INMON, 1999).

Com a operação de slice-dice, torna-se possível analisar os dados de um cubo de através de qualquer uma das dimensões de forma igual. Sendo assim, esta operação permite conhecer em detalhes o relacionamento do valor de uma dimensão com as demais. Considerando ainda o mesmo exemplo do cubo de aquisição de crédito mencionado acima, poderíamos comparar através do slice-dice, os índices de inadimplência de cartões de crédito dos últimos dois anos da região sudeste com os da região nordeste. Através desse conhecimento, poderíamos descobrir variáveis que permitam auxiliar na decisão de ser mais ou menos flexível na oferta de crédito para determinada população.

3.7. Drill-down e Roll-up

Além do slice-dice, temos outras operações muito interessantes nas análises dimensionais, através das quais se torna possível analisar o detalhamento ou a sumarização dos dados no DW. Estamos falando de drill-up e roll-down, e podemos observar um exemplo dessas operações através de um dado de localidade na figura 13.

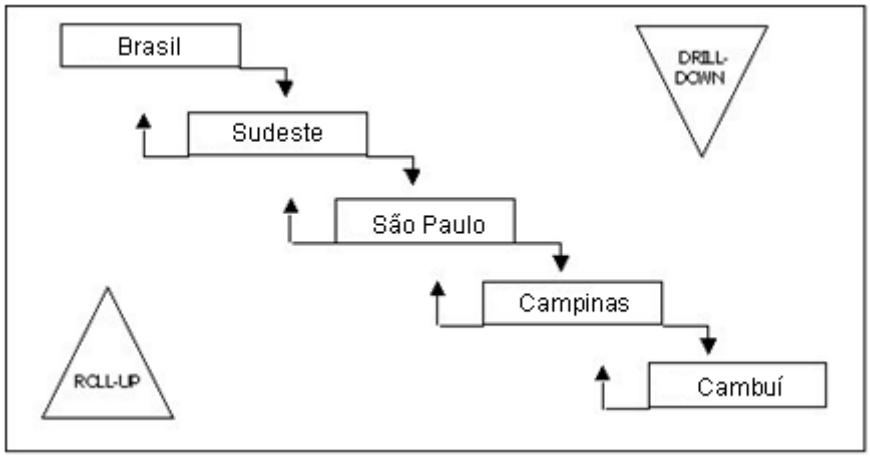

Figura 13 – Exemplo: drill-down e roll-up aplicados em dados de localidade.

A operação de drill-down consiste em consultas mais detalhadas e restritas, envolvendo mais detalhes nos critérios de seleção, enquanto o roll-up é justamente o contrário. Sendo assim, o usuário pode iniciar um estudo em um nível mais alto de agregação e ir aprofundando a análise através te diversos níveis, até atingir o nível mais alto de detalhamento.

Há ainda um tipo de drill-down específico conhecido como drill-across, que é usado para mostrar o relacionamento de resumo entre as diferentes instâncias de dados dentro do ambiente OLAP, é também conhecido como drill-across. Dados mais detalhados existem no nível estruturado organizacional do data warehouse, o que dá suporte a um nível de drill-down que vai além do projeto de cada instância de OLAP departamental (INMON, 1999).

4. ARQUITETURA DO DATA WAREHOUSE

A melhor arquitetura adotada para o Data Warehouse varia conforme o tipo de assunto abordado, portanto, deve variar de acordo com as necessidades a serem atendidas de cada empresa. A seguir, conheceremos a arquitetura genérica de um DW, e posteriormente a de duas e três camadas.

4.1. Arquitetura genérica do Data Warehouse

Uma arquitetura genérica de Data Warehouse compreende os papéis e funcionalidades sistemáticas de cada componente dentro do ambiente de DW, explorando a estrutura de dados, comunicação, processamento, e geração de informação dentro da empresa. Esta composição genérica por camadas apresentada na figura 14, que geralmente é atendida apenas por um software, permite que diferentes abordagens encontradas hoje no mercado sejam adaptadas a ela.

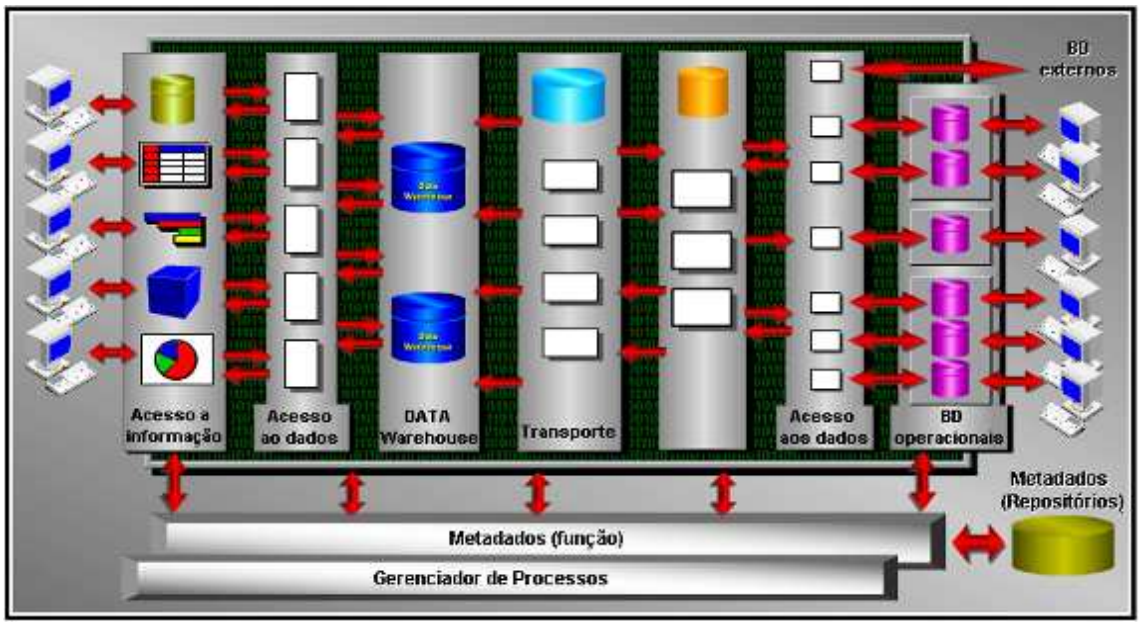

Figura 14 – Arquitetura ambiente de DW.

As camadas que compõem o modelo genérico, conforme observamos na figura 14 são:

- **Camadas de bancos de dados operacionais e fontes externas:** São compostas pelos dados de sistemas operacionais legados das organizações e dados provenientes de fontes externas que serão integrados para compor o DW;
- **Camada de acesso à informação:** Composto pelo conjunto de hardware e o software utilizados para obtenção de relatórios, planilhas, gráficos e consultas, e tem o real objetivo de gerar informações. Nesta camada, os usuários finais interagem com o DW, utilizando ferramentas de manipulação, análise e apresentação dos dados, incluindo-se as ferramentas de Data Mining e visualização;
- **Camada de acesso aos dados:** Esta é a camada de interface entre as ferramentas de acesso à informação e os bancos de dados operacionais. Através dela, se torna possível a comunicação com diferentes sistemas de bancos de dados, sistemas de arquivos e fontes sob diferentes protocolos de comunicação, o que se chama acesso universal de dados;
- **Camada de metadados:** Como já vimos anteriormente, metadados são dados sobre dados, ou seja, são informações que descrevem os dados utilizados pela empresa. Sendo assim, metadados podem ser considerados fórmulas utilizadas para cálculos, descrições das tabelas disponíveis aos usuários, descrições dos campos das tabelas, permissões de acesso, informações sobre os administradores do sistema, entre outros;
- **Camada de gerenciamento de processos:** camada responsável pelo controle das tarefas que mantêm o sistema atualizado e consistente, gerenciando as diversas tarefas que são realizadas durante a construção e a manutenção dos componentes de um sistema de DW;
- **Camada de gerenciamento de replicação:** Esta camada inclui todos os processos necessários para selecionar, editar, resumir, combinar e carregar o DW e as correspondentes informações de acesso a partir das bases operacionais;
- **Camada de transporte:** tem a função de gerenciar a transmissão das informações pelo ambiente de rede que serve de suporte para o sistema como um todo, separando as aplicações operacionais do formato real dos dados, realiza ainda a coleta de mensagens e transações e se encarrega de entregá-las nos locais e nos tempos determinados;

- **Camada do Data Warehouse:** Camada formada pelo armazenamento físico dos dados analíticos provenientes dos sistemas operacionais da empresa e externos. Permite o acesso mais rápido e seguro aos dados do DW, além de prover maior flexibilidade de tratamento e facilidade manipulação.

Resumidamente, a arquitetura genérica compreende a camada dos dados operacionais que serão acessados pela camada de acesso a dados. As camadas de gerenciamento de processos, transporte e data warehouse são responsáveis por distribuir os dados e estão no centro da arquitetura. A camada de acesso à informação possibilita a extração das informações do DW utilizando um conjunto de ferramentas.

4.2. Arquitetura de duas camadas

Uma arquitetura em duas camadas consiste na implantação de sistemas de DW baseada em um computador de alta capacidade como servidor. Este método propõe uma arquitetura que disponibiliza aplicações aos usuários e analistas finais na forma de ferramentas front end, que serão utilizadas para realizar consultas e extrair informações do DW, em conjunto com ferramentas back end, que são as ferramentas responsáveis pela extração, limpeza e carga de dados no DW.

Conforme podemos observar na figura 15, teremos uma arquitetura de DW baseada em duas camadas, o qual possui os componentes dos clientes (front end) e os componentes do servidor (back end).

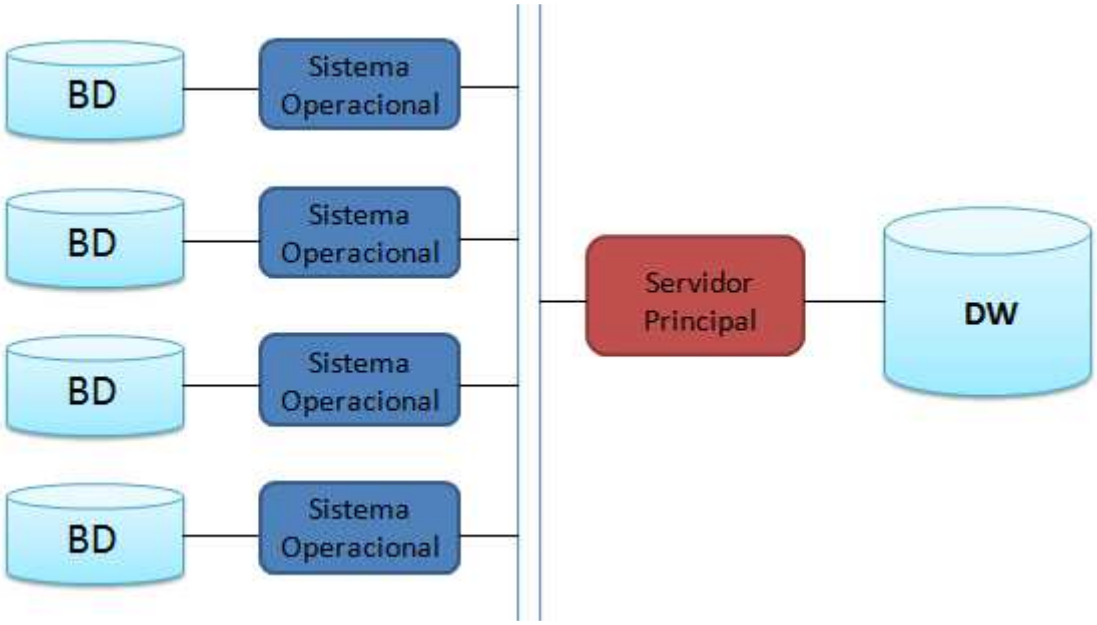

Figura 15 – Arquitetura de duas camadas

Uma das vantagens desta arquitetura, é que ela utiliza os sistemas já existentes na organização bem como os servidores de bancos de dados e requer um pequeno investimento em hardware e software. Por outro lado, ela apresenta a inconveniência da falta de escalonamento, o que resulta, com o aumento do número de usuários, numa performance ruim pelo gargalo existente entre os clientes e o servidor. Esses problemas podem ser freqüentes devido ao uso de estações clientes com performance atenuada e com muitos processos rodando simultaneamente.

Além disso, é comum encontrar no cenário competitivo das grandes a incorporação de outras empresas, onde gradualmente diversos sistemas de computação legados são acumulados, sendo que cada um possui as suas incompatibilidades de definições dos dados. Este processo acaba gerando redundância e falta de consistência dos dados, resultando tanto na dificuldade para administrar as bases de dados quanto para desenvolver novas aplicações front end. Uma alternativa para este problema pode ser a centralização em uma plataforma única baseada na arquitetura de duas camadas, com a finalidade de facilitar o acesso aos dados em um único servidor principal.

4.3. Arquitetura de Três Camadas

A arquitetura de três camadas é uma alternativa para resolver problemas de performance resultantes do gargalo da arquitetura de duas camadas. É uma arquitetura amplamente utilizada pelas empresas, ao passo que oferece uma estrutura bastante flexível e suporta um grande número de serviços integrados, onde a interface do usuário (ferramentas front end), as funções de processamento do negócio e as funções de gerenciamento do BD são separadas em processos, que podem ser distribuídos através dos seus componentes.

Conforme podemos observar na figura 16, na primeira camada fica as aplicações de interface com os usuários, que devem ser gráficas, amigáveis e baseadas em rede. Na segunda camada, conhecida como camada central, encontramos um servidor de alta velocidade onde ficam armazenados os dados e regras de negócio que podem ser compartilhados pela organização, assim como o banco de dados para o DW. E por último, na terceira camada estão localizadas as fontes de dados.

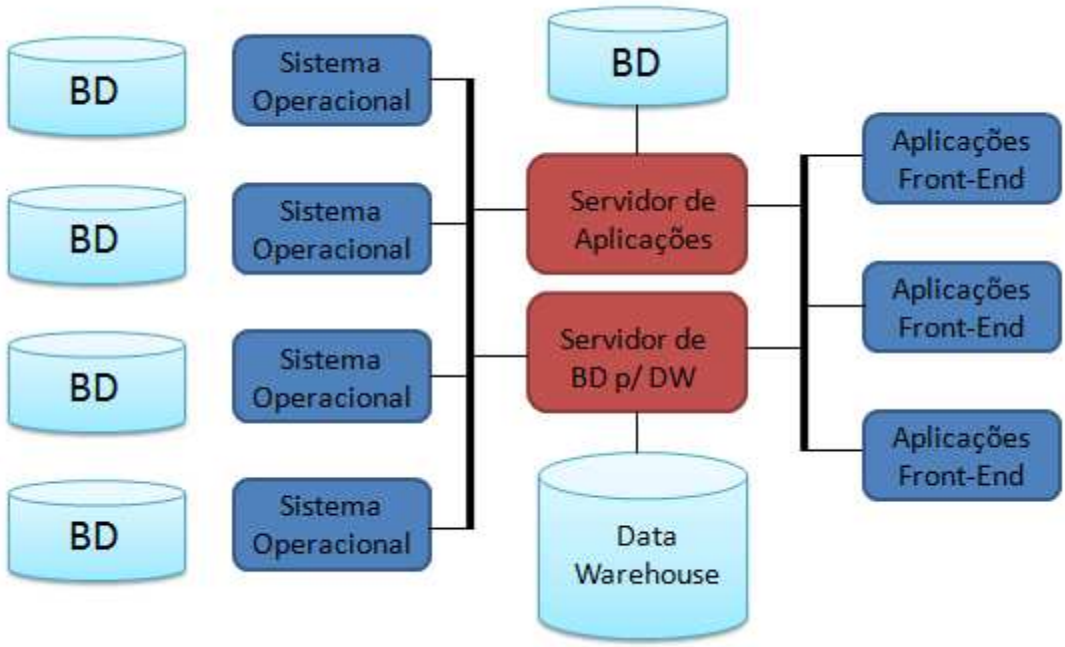

Figura 16 – Arquitetura de três camadas.

Se por um lado a arquitetura de três camadas oferece escalabilidade e maior performance em consultas e processamento, por outro, requer disponibilidade de equipamentos e recursos satisfatórios de conexão entre os diversos componentes do sistema, o que acaba elevando o custo do projeto. É importante frisar, que não existe uma arquitetura universalmente correta. Cada empresa deve considerar suas respectivas necessidades, requisitos e objetivos estratégicos, para selecionar a arquitetura mais viável que atenda e satisfaça o seu negócio.

5. FERRAMENTAS DE BACK END – ETL

Os dados que compõem o ambiente de Data Warehouse, são inicialmente extraídos de um ambiente operacional e de fontes externas, e posteriormente sofrem alguns processos de transformação, que envolve uma série de procedimentos como limpeza, combinação, eliminação, entre outros. Por fim os dados chegam com uma qualidade adequada ao ambiente de Data Warehouse através de processos de carga de dados, e estão disponíveis para estudos e análises que posteriormente apoiarão tomadas de decisão. Portanto, o conjunto destes procedimentos desde a extração de dados do ambiente operacional até a carga de dados no Data Warehouse compõe o processo de ETL, conforme podemos observar na figura 17. Neste capítulo, exploraremos um pouco mais de cada uma dessas etapas.

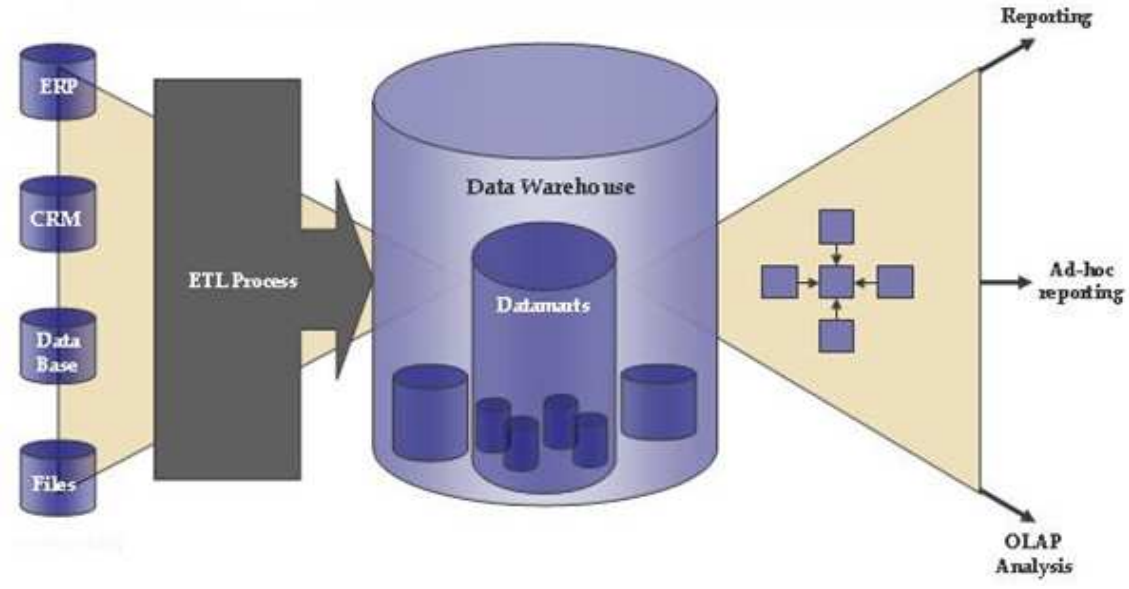

Figura 17 – Extração, transformação e carga de dados no ambiente de Data Warehouse.

5.1. Staging Area

Como já mencionado acima, antes dos dados estarem devidamente disponíveis no DW, existe uma etapa que contempla a movimentação dos dados para entre os sistemas operacionais e o DW. Essa movimentação se dá basicamente em três passos: a extração, a transformação e a carga dos dados.

De acordo com Kimball (2002), a data staging area abrange tudo entre os sistemas operacionais de origem e a área de apresentação dos dados. Portanto, podemos considerar a staging área como uma área de armazenamento e conjunto de processos que limpam, transforma, combinam, retiram duplicidades, armazenam e preparam dados de origem para serem utilizados no data warehouse.

5.2. Extração dos Dados

A primeira fase para a obtenção de dados no ambiente de Data Warehouse é a extração de dados. É comum que os dados sejam extraídos de diversas fontes de dados, incluindo bancos de dados transacionais, planilhas de Excel, arquivos de massa de dados, entre outros.

A etapa de extração de dados significa basicamente ler e entender as fontes de dados e copiar as partes necessárias para a área de transformação de dados, a fim de serem trabalhadas posteriormente (KIMBALL, 2002).

Esta etapa do projeto de um DW pode ser considerada bastante exaustiva ao passo que contempla a análise dos requisitos de dados e informações necessários para os processos de negócio e que deverão ser suportados pelo DW. Portanto, é nesta etapa que ocorre a análise de todas as fontes dos dados, de forma a compreender, integrar e refletir uma perspectiva histórica de interesse as análises.

A extração de dados pode ser realizada através de programas com códigos de rotina de execução que geram arquivos contendo os dados desejados. Outra alternativa é utilizar programas e ferramentas específicas de extração, que já geram o código próprio e disponibilizam os arquivos em formato padrão e não proprietário.

5.3. Transformação

Após os dados serem devidamente extraídos dos sistemas fontes, passam por um conjunto de processos de transformação, de modo a convertê-los para que possam adquirir um formato adequado para carga e posteriores análises que auxiliarão processos decisórios.

As rotinas de limpeza e integração atuam sobre os dados extraídos. A execução dessas rotinas de limpeza sobre os dados coletados permite assegurar sua consistência. Dados migrados para o DW, sem integração, não podem ser empregados no suporte a uma visão corporativa dos dados (INMON, 1997).

Segundo Kimball (2002), as características mais relevantes para garantir a qualidade dos dados são:

- Unicidade, evitando assim duplicações de informação;
- Precisão. Os dados não podem perder suas características originais assim que são carregados para o DW;
- Completude, não gerando dados parciais de todo o conjunto relevante às análises;
- Consistência, ou seja, os fatos devem apresentar consistência com as dimensões.

Sendo assim, dentre os processos da etapa de transformação mais comumente utilizados, podemos considerar:

- Limpeza: A finalidade deste processo é deixar os dados com formatos padrões, corretos, consistentes e não duplicados, tornando-os assim o mais próximo possível da realidade. Consiste em um conjunto de atividades como correção de erros de digitação, descoberta de violações de integridade, substituição de caracteres desconhecidos, a padronização de abreviações, entre outros.
- Eliminação: consiste apenas em desconsiderar um conjunto de dados dos sistemas fontes, que não serão úteis para o DW.
- Combinação: Ocorre quando existe duas ou mais fontes de dados que possuem valores de chaves representando registros iguais ou complementares.

- Desnormalização e normalização: o padrão no processo de transformação é reunir as hierarquias de dados, separadas em várias tabelas devido à normalização, dentro de uma única dimensão, de forma desnormalizada. Pode ocorrer, entretanto, que provenientes do processo de extração estejam completamente desnormalizados dentro de arquivos texto, nesse caso é possível que seja necessário normalizar partes dos registros (HOKAMA, 2004).
- Cálculos, derivação e alocação: Consiste em algumas transformações aplicadas às regras do negócio identificadas durante o processo de levantamento de requisitos. Normalmente, as ferramentas a serem empregadas possuem um conjunto de funções, tais como manipulação de textos, aritmética de data e hora, entre outras (HOKAMA, 2004).

5.4. Carga no Data Warehouse

Ao serem extraídos e transformados os dados agora já possuem um valor adequado para serem analisados. Portanto, a próxima etapa consiste na carga dos dados já devidamente tratados no Data Warehouse.

O tipo da carga de dados é um dos fatores que deve ser analisado dependendo da dinâmica do negócio em questão. Pode ser considerada a opção de carga total, comum para tabelas de dimensão, onde os dados serão completamente excluídos, e posteriormente os atuais serão inseridos. Outra alternativa, na maioria das vezes mais viável, é a carga do tipo incremental, comum para tabelas de fato, onde apenas serão inseridos os registros que compõem as alterações em relação à última atualização.

Ao efetuar a carga, é importante garantir que haja correspondência entre o relacionamento das tabelas que compõem o novo ambiente projetado. Portanto, nesta etapa, torna-se necessário checar integridade entre chaves primárias e secundárias, assegurando um ambiente analítico integro e confiável.

6. EXTRAÇÃO DE INFORMAÇÃO E FERRAMENTAS DE FRONT-END

Até agora, conseguimos focar o conhecimento para absorver algumas definições e aprender sobre as estruturas genéricas que acercam a tecnologia do Data Warehouse. Portanto, chegou a hora de conhecermos o que é o produto do Data Warehouse e um pouco mais sobre o que ele tem a oferecer para o mundo corporativo.

Primeiramente, para que possa ser competitiva, toda organização deve ser capaz de tomar decisões com agilidade e eficácia para que possa atingir seus objetivos e metas.

O planejamento estratégico e as metas gerais da empresa compõem o cenário em que são desenvolvidos os processos que agregam valor e são tomadas as decisões necessárias para a organização atingir vantagens competitivas que a garantam perenidade. Neste contexto, encontramos frequentemente o Data Warehouse, entre outras tecnologias de informação, como ferramenta de suporte ao planejamento estratégico e à tomada de decisões dentro das empresas. É importante frisar, que não existe um modelo padrão que oriente as empresas ou grandes organizações em relação à adoção das diversas tecnologias de informação existentes no mercado. As companhias devem alinhar a estratégia dos sistemas de informação com as estratégias e objetivos gerais dos negócios.

Sendo assim, o objetivo deste capítulo é justamente não ficar preso apenas nas fronteiras de definições e arquitetura de ferramentas de consulta (ferramentas de Front-End), como também extrapolar o conhecimento para a aplicação da ferramenta e como o produto final da tecnologia pode agregar valor no negócio.

6.1. Extração de Informação

Atualmente, muito se fala em tomada de decisão e processos decisórios, mas os grandes desafios quando se aborda esse assunto, frequentemente estão voltados para as decisões não programadas.

Existe um tipo de decisão, as chamadas decisões programadas, que podem ser tomadas através de regras, procedimentos ou métodos quantitativos, justamente

porque envolvem problemas bem estruturados, compostos por fatores de certa forma relacionados, onde a mesma decisão pode ser aplicada inúmeras e repetidas vezes.

Já as decisões não programadas lidam com situações incomuns ou excepcionais. Em muitos casos, essas decisões são difíceis de mensurar ou quantificar, sendo que cada decisão não programada agrega diversas características exclusivas que tornam a aplicação de regras ou procedimentos não triviais ou inválidos.

O Data Warehouse dentro das empresas deve fornecer uma estrutura capaz de prover dados que possam ser facilmente transformados em informação a ser exibida adequadamente, através de uma interface de consulta no formato de relatórios. Portanto, a tecnologia que envolve o Data Warehouse, através de MISs e DSSs, deve servir tanto como suporte para decisões programadas, ao passo que fornece informações necessárias para descrever, por exemplo, problemas de rotina, como também para decisões não programadas, que envolvem problemas não estruturados e que não possuem uma solução exata, admitindo assim mais de uma alternativa.

6.1.1. MIS (Management Information System)

Através das ferramentas de consulta chamadas ferramentas de Front-End, cuja arquitetura será mais bem detalhada nas próximas páginas ainda deste capítulo, torna-se possível efetuar consultas organizadas no Data Warehouse e gerar relatórios mais conhecidos como MIS, que oferecem percepções detalhadas aos administradores e gerentes a respeito de operações diárias da organização. Essa percepção detalhada permite que os administradores da empresa controlem, organizem e planejem as atividades operacionais em mais detalhe e de modo mais eficiente. É imprescindível que os MIS sejam capazes de fornecer as informações certas à pessoa certa, de modo adequado e na hora correta.

Como já vimos algumas vezes nos capítulos anteriores, os dados que compõem o Data Warehouse são provenientes de diversos sistemas operacionais ou transacionais, além de fontes externas de dados. Portanto, a seguir,

conheceremos alguns exemplos de formatos de saída que alguns relatórios ou MIS podem assumir e como estes podem ser úteis dentro da companhia:

- **Relatórios Agendados**: São produzidos periodicamente ou de acordo com alguma política de agendamento programada, como por exemplo, diariamente, semanalmente ou mensalmente. Como exemplo, um gerente de uma área de produção, poderia solicitar um relatório semanal relacionando todos os gastos feitos pela empresa com folha de pagamento e despesas diretas de fabricação, com o intuito de controlar os custos incorridos pelos funcionários e outros fatores de produção. Outros relatórios agendados podem ajudar administradores e gerentes a controlar o crédito de seus clientes, o desempenho de um representante de vendas, estoque e muito mais. Ainda, os relatórios agendados, podem ser chamados de relatório de indicador-chave, quando estes resumem as atividades críticas do dia anterior e geralmente são disponibilizados no início de cada dia de trabalho. Como exemplo, podemos citar um relatório indicador-chave diário que contemple índices de inadimplência de uma carteira de clientes de uma instituição financeira, com o intuito de acompanhar as rolagens de inadimplência ao longo do mês. Os relatórios de indicadores-chave são utilizados por gerentes e executivos para a tomada de ações corretivas rápidas com relação a aspectos relevantes ao negócio em pauta (STAIR, 2010).

- **Relatórios sob Demanda**: são relatórios desenvolvidos para fornecimento de informações não padrão e são elaborados sob requisição. Como exemplo, um executivo poderia solicitar um relatório que contemplasse estágios de produção em que cada um dos pedidos se encontra. Da mesma forma, um gerente ou diretor de uma instituição financeira poderia estar interessado em saber qual é o valor exato de recebíveis de um determinado produto de crédito em uma determinada época do ano ou do mês.

- **Relatórios de Exceções:** são produzidos automaticamente sempre que uma situação incomum acontece ou que requeira atenção da administração. Os relatórios de exceção são usados frequentemente para o monitoramento de aspectos cruciais ao sucesso da empresa que os aplica. Portanto, quando um relatório de exceção é produzido, um administrador ou executivo toma alguma espécie de providencia. Os

parâmetros de disparo para geração dos relatórios (triggers) devem ser ajustados delicadamente. Em caso de um ponto de disparo muito baixo, relatórios serão gerados de forma excessiva. No caso de pontos de disparo muito alto, casos problemáticos podem passar despercebidos. Um exemplo para este caso, considerando ainda uma instituição financeira, pode ser um relatório gerado automaticamente quando ocorrem diversos casos de fraude de produtos de crédito de uma mesma modalidade ou em um local específico. Neste caso, isso poderia caracterizar um cenário de ataque de fraude, onde ações rápidas devem ser tomadas para que futuros ataques sejam impedidos.

- **Relatórios Detalhados:** fornecem informações e resultados em diversos níveis de detalhes. Isso se torna possível com uso dos recursos de roll-up e drill-down, facilitando assim a análise e compreensão mais precisas da organização em questão.

Além de assumir estes formatos, os relatórios ou MISs ainda devem ser direcionados e divididos entre as linhas funcionais da empresa como finanças, marketing, produção, entre outros. Abaixo, segue alguns exemplos de como os dados e informações extraídos do Data Warehouse são aplicados e podem atender algumas áreas funcionais dentro da organização:

- **Área Financeira:** no geral, são gerados relatórios que integram as informações financeiras e operacionais, monitoram e controlam o emprego de recursos no tempo. Portanto, são capazes de auxiliar na aquisição, uso e controle de dinheiro, fundos e outros recursos financeiros. As empresas que não são capazes de administrar e empregar seus fundos de maneira eficiente, muitas vezes sofrem com lucros baixos ou até mesmo vão à falência. Exemplos de usos internos de fundos incluem reposições de estoque, construções e atualizações de instalações e equipamentos, contratações de horas de trabalho adicionais, aquisição de outras empresas, compra de matéria-prima, entre outros.

Além disso, MISs financeiros devem fornecer informações a indivíduos e grupos externos, incluindo acionistas, agências federais e auditorias internas ou externas.

Com dados provenientes do DW, ainda torna-se possível gerar relatórios de perdas e ganhos, conhecidos como P&L (Profits & Losses).

- **Produção:** através de relatórios que monitoram e controlam o fluxo de materiais, produtos e serviços dentro de uma organização. Podem auxiliar diretamente no controle de estoque, no escalonamento da produção nas instalações de manufatura, controle de processos e de qualidade, além de auxiliar na previsão e estimativa de demandas futuras e presentes de novos produtos ou serviços.

- **Risco de crédito:** é possível gerar relatórios de controle e monitoramento de inadimplência e fraude, acompanhamento de políticas e estratégias de aquisição e manutenção de produtos de crédito. Assim, torna-se possível, por exemplo, rentabilizar mais a carteira, ofertando mais produtos para clientes com perfil menos arriscado e diminuindo a perda com políticas de cobrança mais rígidas para clientes que apresentam um comportamento mais arriscado e propenso a inadimplência.

- **Marketing:** através de relatórios que dão suporte a atividades administrativas nas áreas de desenvolvimento de produtos, decisões de preços, eficiência promocional e previsão de vendas. O Data Warehouse combinado com outras ferramentas de BI e técnicas de Data Mining, compõem mecanismos imprescindíveis para o departamento de marketing.

6.1.2. DSS (Decision Support System)

Conforme colocado no início deste capítulo, os dados disponíveis no Data Warehouse, devem ser capazes de apoiar também as decisões não programadas, que representam problemas específicos e não estruturados. Este processo torna-se possível através dos DSSs ou SADs (traduzindo - Sistemas de Apoio a decisão). Os SAD podem trazer aumentos na lucratividade, redução nos custos e melhores produtos e serviços.

O SAD é uma ferramenta utilizada mais frequentemente por profissionais de cargos executivos mais altos. Até certo ponto, gerente de todos os níveis tem de

lidar com problemas não estruturados e pouco usuais, porém a quantidade e o impacto das decisões tomadas aumentam conforme aumenta o nível do gerente ou administrador na hierarquia da empresa. Portanto, o SAD é utilizado como uma forma de trazer um pouco mais de estrutura para estes tipos de problemas e auxiliar na tomada de decisão (STAIR, 2010).

Ao permitir todos os tipos de abordagens para tomada de decisão, um SAD dá ao responsável pelas decisões bastante flexibilidade no que diz respeito ao apoio computacional recebido. Assim, através do SAD é possível realizar uma análise do tipo e-se (what-if analysis), que é um processo pelo meio do qual são feitas mudanças hipotéticas no conjunto de dados de um problema e observados os impactos em seu resultado. Por exemplo, dado problemas de demanda de produtos como automóveis, torna-se possível determinar as peças e os componentes em estoque necessários para atendê-la. Basta o administrador fazer mudanças no conjunto de dados do problema (neste caso, número de automóveis que devem ser produzidos para o mês seguinte) e obter imediatamente qual será o impacto na mudança nos requisitos por peça. Outra forma flexibilidade presente no SAD, é a possibilidade da tomada de decisão multicriteriada, que faz com que gerentes e administradores considerem diversos objetivos e metas (STAIR, 2010).

Além de flexível, o SAD tem a capacidade de simular diversas situações e cenários. Podemos, por exemplo, realizar uma análise em busca de objetivos (goal-seeking analysis), que é o processo por meio do qual os dados de um problema são determinados para que seja possível atingir um resultado específico pré-determinado.

6.2. Ferramentas de Front-End

Agora que já temos conceitos sobre a construção física e lógica do Data Warehouse, nos resta conhecer um pouco sobre as ferramentas de Front-End. Estas ferramentas, nada mais são do que ferramentas de consulta que interagem e fazem a interface com o gerenciador de banco de dados do Data Warehouse, enviando solicitações SQL e exibindo o resultado formatado em modo texto ou modo gráfico.

É através dela que o usuário final acessa o DW, efetua consultas e gera relatórios que darão suporte a decisões.

6.2.1. Arquitetura Interna de Ferramentas de Consulta

Como sabemos, o SQL é a linguagem padrão para comunicação com SGBDs, e através dela conseguimos efetuar consultas que retornam um número de respostas relativamente pequeno. Porém, na maioria dos casos, esse conjunto pequeno de respostas não atende a uma pergunta abrangente do negócio. Sendo assim, uma pergunta abrangente do negócio exige um relatório de negócio, onde são apresentados vários tipos de informações simultaneamente e também diversas comparações. Para fim de qualquer tipo de análise, é comum obtermos referencias para comparação, como por exemplo, a receita de vendas ou mesmo as perdas deste mês comparadas com as do mês anterior.

Além das comparações, há uma série de formatos e modos de apresentação que afeta a maneira pela qual os dados são exibidos para o usuário final, como por exemplo, destaque, intermitência, exibição ou não de linhas inteiras dependendo de seus valores, classificação, média móvel, soma móvel, entre outros. Como a grande maioria destes modos não são fornecidos em SQL padrão, é função da ferramenta de consulta um pós-processamento do conjunto de respostas (KIMBALL, 2002).

Considerando o relatório de negócio da figura 18 apenas como exemplo:

Produto	Região	Vendas no mês	Crescimento nas Vendas versus mês anterior	Vendas como % da categoria	Mudança nas vendas como % da Categoria versus mês anterior	Mudança nas vendas como % da Categoria versus ano anterior
X	Central	110	**12%	31%	3%	7%
X	Leste	179	-3%	28%	-1%	3%
X	Oeste	55	5%	**44%	1%	5%
Total X		344	**6%	33%	1%	5%
Y	Central	66	2%	18%	2%	**10%
Y	Leste	102	4%	12%	5%	**13%
Y	Oeste	39	-9%	9%	-1%	8%
Total Y		207	1%	13%	4%	**11%
Total Geral		551	4%	20%	2%	8%

Figura 18 – Exemplo: relatório de negócio.

Em relação à primeira etapa do pós-processamento, referente às comparações, a ferramenta de Front-End gera cada componente de uma comparação com uma consulta separada e por final, são unidas as respostas.

6.2.2. Interface de Usuário

Uma interface de usuário ideal deve basear-se em reconhecer e apontar, e não em relembrar e digitar. Uma interface de usuário ideal minimiza o número de cliques de botão e mudanças de contextos (KIMBALL, 2002).

Em termos de visibilidade, uma ferramenta de Front-End deve exibir de imediato ou com um único clique de botão, os principais itens de relance, como:

- As dimensões;
- As restrições escolhidas nessas dimensões;
- A tabela de fatos de nível básico;
- O estado atual do relatório.

Não seria possível e nem desejável exibir os elementos de dados de cada uma das tabelas e muito menos a listagem de atributos dimensionais para cada dimensão, mas por outro lado, o ideal seria que a ferramenta orientasse o usuário em relação ao contexto geral do relatório que está sendo gerado. Assim, seria pertinente que o usuário pudesse visualizar todos os itens que estão sendo utilizados na tela de uma só vez, assim como visualizar de forma sumarizada todas as tabelas disponíveis, e que com apenas um clique, seja possível exibir uma dimensão completa.

Uma ferramenta de consulta para gerar relatórios deve comunicar o contexto do relatório imediatamente, incluindo especialmente a identidade das tabelas de dimensão, da tabela de fatos, as restrições em vigor nas dimensões e o estado atual do relatório. O relatório propriamente dito deve ficar visível com um único clique de botão (KIMBALL, 2002).

Caso os usuários possam visualizar algumas coisas, como uma coluna de um relatório, deve ser possível editá-la diretamente. Uma boa ferramenta de consulta

permitirá que o usuário mude, por exemplo, uma restrição em uma coluna com um simples clique.

Outro recurso simples e interessante é o comando de parar imediatamente a consulta, uma vez que o usuário pode mudar de idéia ou perceber que ouve um erro na formulação da consulta.

Como último ponto, mas não menos importante, uma boa ferramenta de consulta deve suportar pelo menos três modos de utilização ou de interface:

- Interface de executivo: tem como objetivo principal alertar o executivo para algo fora do usual em um relatório predefinido. A interface do executivo deve ser capaz de destacar automaticamente alguns itens-chave e esperar que o usuário selecione um deles e solicite um nível de maior detalhe.
- Interface de analista: destina-se principalmente a gerar e modificar relatórios predefinidos e parametrizados. Ainda está dentro do escopo deste tipo de interface modificar os modos de apresentação de colunas e adicionar novas especificações predefinidas ao relatório.
- Interface de desenvolvedor: tem como principal objetivo construir relatórios predefinidos e parametrizados. O desenvolvedor deve ser capaz de ver todas as tabelas de fatos e de dimensão do Data Warehouse e de combinar partes delas em um modelo de relatório predefinido.

6.2.3. Recursos

Além de recursos básicos e triviais desejáveis em uma ferramenta de Front-End como, por exemplo, browsing, interface de ajuda, funções de comparações e cálculos distribuídos, apresentaremos a seguir alguns interessantes recursos adicionais.

6.2.3.1. Efetuando o Drill-Down

Segundo Kimball (2002), drill-down não significa descer em uma hierarquia predeterminada. Significa a possibilidade de obter rapidamente cabeçalhos de linha

de qualquer uma das dimensões associadas a uma tabela de fatos. Significa também a possibilidade de remover cabeçalhos e pesquisar em direções diferentes.

A primeira reação que um executivo ou tomador de decisão tem ao observar um relatório é perguntar por quê. Na realidade, isso significa especificamente “Forneça detalhes mais interessantes referentes a esse item”. Sendo assim, drill-down não representa somente descer na hierarquia. Considere o exemplo da figura 19:

Produto	Região	Vendas no mês	Crescimento nas Vendas versus mês anterior
X	Central	110	**12%
X	Leste	179	-3%
X	Oeste	55	5%
Total X		344	**6%

Figura 19 – Exemplo: Drill-Down

Poderíamos efetuar um drill-down incluindo o atributo de tamanho do produto no cabeçalho, mas não necessariamente notaríamos um padrão revelador que nos indicasse o porquê da queda nas vendas do produto X na região Leste. Sendo assim, mesmo que houvesse um nível a mais de detalhe, os diversos tamanhos do produto poderiam estar sendo afetados uniformemente, inutilizando o detalhamento naquela dimensão.

Portanto, em vez de criarmos seleções de atributos automáticas, podemos arrastar oportunamente qualquer atributo que nos pareça mais plausível. Neste caso, explorando a suspeita na força de vendas, podemos encontrar:

Produto	Região	Equipe de Vendas	Vendas no mês	Crescimento nas Vendas versus mês anterior
X	Central	Chicago	52	**21%
X	Central	St. Louis	28	5%
X	Central	Dallas	30	6%
X	Total Central		110	**12%
X	Leste	Nova York	93	4%
X	Leste	Boston	75	5%
X	Leste	Washg	11	-15%
X	Total Leste		179	-3%
X	Oeste	Los Angeles	18	5%
X	Oeste	São Franc	16	4%
X	Oeste	Seattle	21	6%
X	Total Oeste		55	5%
Total X			344	**6%

Figura 20 – Utilizando o Drill-Down

Agora sim, conseguimos responder o porquê inicial e caminhar de através do drill-down de forma promissora. Ao entrarmos no detalhe da força de vendas, percebemos uma discrepância entre as equipes. Podemos notar que o resultado surpreendente da região Central deve-se a Chicago, e a queda de vendas significativa na região Leste deve-se ao fraco desempenho de vendas da equipe de Washington.

Além de modificar os cabeçalhos do relatório, efetuando o Drill-Down, ainda podemos combinar duas tabelas de fatos que compartilhem dimensões, efetuando o Drill-Across, caso quisermos, por exemplo, uma cadeia de valor. Conforme exemplo citado por Kimball (2002), se tivermos as remessas de um fabricante para um grupo de lojas e mais adiante na cadeia de valor tivermos as vendas das lojas, as tabelas de fatos expedições do fabricante e de vendas da loja provavelmente terão as seguintes dimensões:

Expedições
Tempo
Produto
Loja
Acordo do Fabricante
Transportadora

Vendas
Tempo
Produto
Loja
Promoçao Consumidor
Transportadora

Figura 21 – Exemplo: Drill-Across - tabelas de fatos.

Fica fácil visualizarmos como o drill-across funciona, se imaginarmos dois relatórios usando os mesmos cabeçalhos de linhas das duas tabelas de fatos:

Produto	Semana	Quantidade Expedições
X	07/nov/11	66
X	14/nov/11	76
X	21/nov/11	56

Produto	Semana	Quantidade Vendida
X	07/nov/11	62
X	14/nov/11	63
X	21/nov/11	74

Figura 22 – Exemplo: Drill-Across - relatórios de expedições e de vendas respectivamente.

Assim fica intuitivo imaginar que podemos combinar esses dois relatórios em um único demonstrado na figura 23 utilizando drill-across.

Produto	Semana	Quantidade Expedições	Quantidade Vendida
X	07/nov/11	66	62
X	14/nov/11	76	63
X	21/nov/11	56	74

Figura 23 – Exemplo: Drill-Across - relatórios final drill-across

Este é apenas um exemplo simples, mas poderíamos ainda, por exemplo, criar colunas de comparação interessantes entre a quantidade expedida e a quantidade vendida, como o excedente em estoque.

Segundo Kimball (2002), será possível construir um relatório drill-across contanto que os cabeçalhos de linha escolhidas tenham exatamente o mesmo significado em todas as tabelas de fatos e dimensão envolvidas. Será possível aplicar restrições a dimensões não compartilhadas em uma ou mais tabelas de fatos, mas será prudente advertir o usuário sobre essa condição durante a execução ou no relatório propriamente dito.

6.2.3.2. Restrições de Comportamento

Uma tarefa bastante usual, porém de certa forma difícil de ser realizada, é o rastreamento por comportamento. Através dos dados hospedados no data warehouse, torna-se possível identificar e definir grupos ou populações com padrões de comportamento semelhantes e estudá-los de alguma forma. Como exemplo mencionado por Kimball (2002), poderíamos definir um grupo de consumidores em um ambiente varejista que gastou mais de R$100 em suas lojas no mês passado. Sendo assim, esse grupo seria definido unicamente por seu comportamento de compra, e não existe nenhum outro atributo na dimensão cliente para filtrar ou identificar esse grupo. Precisamos identificar esse grupo para posteriormente efetuarmos repetitivas consultas e estudos. Como se comportam em março os grandes consumidores de fevereiro? Há quanto tempo os grandes consumidores de fevereiro são clientes e qual é a média mensal de compras? Quantos dos grandes consumidores de fevereiro também responderam as malas diretas recentes?

Poderíamos gerar um relatório complexo contendo todas essas informações, porém não conseguiríamos capturar esse mesmo grupo para reutilizá-los.

Devido à falta de qualquer atributo para reunir o grupo comportamental, podemos construir uma tabela especial chamada Grandes_Clientes_Fevereiro composta apenas por chaves desse grupo de clientes. Assim, poderemos utilizar a mesma tabela especial em diversas análises e estudos a respeito do comportamento desta população, e ainda encontrar por exemplo, a intersecção, a união ou a diferença entre Grandes_Clientes_Janeiro e Grandes_Clientes_Fevereiro.

Este tipo de recurso geralmente está adicionado na versão da ferramenta de consulta do desenvolvedor e é utilizado e grande parte das análises para estudos de comportamento.

6.2.3.3. Rotacionando

Segundo Kimball (2002), rotacionar é uma função útil para reorganizar linhas e colunas de um relatório depois que ele tiver sido recuperado do DBMS. O SQL

apenas entrega os dados de forma muito peculiar e definida. O conjunto de respostas é retornado como um conjunto de linhas em que todos os itens de uma determinada coluna possuem a mesma definição, e os dados subjacentes são agregados unicamente por combinações do cabeçalho de linha. A rotação permite que cabeçalhos de linha e coluna sejam misturados em combinações arbitrárias. Isso tem como efeito a reorganização dos dados principais.

6.2.3.4. Estendendo Operações SQL

A ferramenta de consulta também deve ser capaz de oferecer ao usuário funções específicas que facilitam as consultas e ainda não estão disponíveis como extensões SQL. Funções estas, que executem em um único passo operações que o usuário levaria duas ou mais etapas para realizar. Como exemplo, podemos colocar as funções PERIODAVG e COUNTAVG, onde a primeira fornece a média correta de períodos de uma medição dividindo-a pela cardinalidade da restrição, e a segunda retorna essa cardinalidade. Assim, seria fácil, por exemplo, saber se as vendas nesse trimestre estão ultrapassando a média de vendas ao longo do ano. Bastaria comparar PERIODAVG(vendas) para o trimestre e PERIODAVG(vendas) para o ano. Os dois números seriam comparáveis porque ambos seriam expressos no nível de detalhe da medição tempo, seja diário, semanal ou mensal.

7. DATA MINING

Conforme já visto nos capítulos anteriores, a proposta do data warehouse é sustentar a tomada de decisão com dados. Porém, a data mining, ou mineração de dados pode ser usada em conjunto com o data warehouse para auxiliar certos tipos de decisão e obtenção de novos padrões ou tendências úteis que não poderiam ser encontrados simplesmente pesquisando ou processando dados no data warehouse.

Data mining se refere à descoberta de novas informações em função de padrões ou regras em grandes quantidades de dados. Geralmente, o sucesso das

aplicações de data mining, dependerá primeiro de uma construção estruturada do data warehouse.

Apenas para contextualizar melhor a ferramenta, convém dizer que a mineração de dados é uma parte do processo de descoberta de conhecimento, conhecido como Knowledge Discovery in Databases (normalmente abreviado em KDD). Conforme podemos observar na figura 24, o processo de KDD é composto pelas seguintes fases: seleção de dados, limpeza, enriquecimento, transformação, data mining e apresentação e interpretação da informação e padrões descobertos.

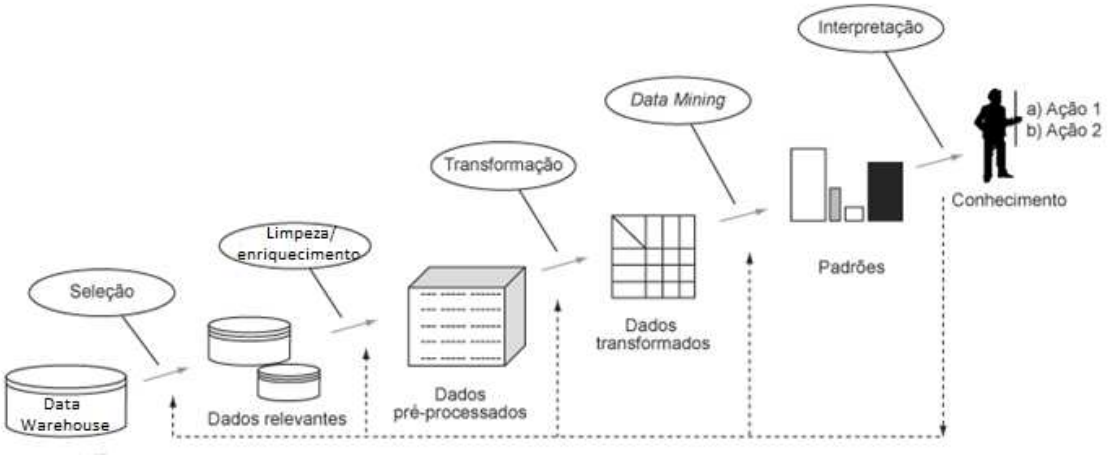

Figura 24 – Etapas do processo de KDD.

Se tomarmos como exemplo um data warehouse de um comerciante varejista, podemos encontrar dados como nome, CPF, CEP, endereço, telefone, loja, data de compra, valor total de compra, quantidade de itens comprados, preço dos itens, entre outros. Durante a etapa de seleção de dados, serão selecionados, por exemplo, apenas itens específicos ou lojas em determinadas regiões do país. Na etapa de limpeza de dados, pode ocorrer a correção de alguns dados, como o formato do CEP, do CFP ou do telefone do cliente que pode estar precedido equivocadamente com prefixo de localidade. Durante o processo de enriquecimento, os registros podem ser incrementados com dados de renda, de risco de crédito ou idade através do CPF. A etapa de transformação de dados deve diminuir o volume de dados, por exemplo, agrupando os diversos produtos em categorias como eletrônicos, alimentos, bebidas, entre outros, de forma a facilitar o processo e a etapa de interpretação final das informações. Os CEPs também podem ser agrupados por regiões geográficas.

Como geralmente o processo de KDD se baseia em dados extraídos de um data warehouse, espera-se que a etapa de limpeza de dados já tenha sido executada anteriormente no processo de ETL. Somente depois dos dados pré-processados, transformados e devidamente preparados é que as técnicas de data mining serão utilizadas para extrair diversas regras e padrões que influenciarão as decisões do negócio.

Considerando o mesmo exemplo do comerciante varejista, teremos como possíveis resultados da aplicação de data mining, a descoberta dos seguintes tipos informações:

- Regras de associação: por exemplo, um cliente que compra macarrão, pode também comprar molho de tomate, queijo ralado ou vinho.
- Padrões seqüenciais: suponha, por exemplo, um cliente que compra uma impressora, e que dentro de três ou quatro meses, ele volte a comprar mais blocos de papel sulfite, de forma que, dentro dos próximos seis meses ele deverá comprar cartuchos de tinta. Um cliente que compra mais de duas vezes num período de baixa temporada estará mais propenso a comprar em períodos sazonais como Natal.
- Árvores de classificação: por exemplo, podemos classificar os clientes por freqüência de visita, por forma de pagamento e por volume de compra ou afinidade por tipos de itens.

Como podemos ver, existem inúmeras possibilidades de descoberta de novo conhecimento sobre padrões de compra relacionando fatores como idade, renda, local de residência, com o que e como muitos clientes compram. Essas informações podem ser muito importantes para planejar, por exemplo, a localização de uma nova loja, promoções nas lojas, para combinar itens nas propagandas ou para planejar estratégias sazonais de marketing.

Segundo Navathe (2005), o processo de data mining é executado levando em consideração um propósito ou meta, que pode ser classificado em uma das seguintes classes:

- Predição: a data mining pode mostrar como certos atributos de dados irão se comportar no futuro.

- Identificação: padrões de dados podem ser usados para identificar a existência de um item, um evento ou uma atividade.
- Classificação: a data mining pode particionar os dados, podendo as diferentes classes ou categorias ser identificadas baseadas em combinações de parâmetros.
- Otimização: o objetivo da mineração de dados pode ser otimizar o uso de recursos limitados como tempo, espaço, dinheiro ou materiais, e maximizar as variáveis de saída como vendas ou lucros sob determinado conjunto de restrições.

A seguir, veremos com um pouco mais de detalhes os tipos de descoberta de conhecimento durante a aplicação do data mining e como eles podem auxiliar a descoberta de conhecimento e de formas de fazer negócio.

7.1. Tipos de descoberta de conhecimento durante a Data Mining

O data mining está estruturado e apoiado, comumente, em uma gama muito ampla de disciplinas, incluindo análises estatísticas e otimização restritiva, assim como aprendizado por máquinas e redes neurais. Não existe nenhum tipo de fronteira separando estas disciplinas, e não está no escopo deste capítulo explorar e discutir em detalhe a gama de aplicações e disciplinas que formam este vasto campo de trabalho. Portanto, aproveitaremos este capítulo para fazer uma maior abordagem sobre os tipos de conhecimento resultantes da aplicação da mineração de dados.

Primeiramente, segundo Navathe (2005), o conhecimento é comumente interpretado como o envolvimento de algum grau de inteligência. Existe uma progressão do dado para a informação e para o conhecimento á medida que evoluímos com o processamento. O conhecimento é classificado em indutivo e dedutivo. O conhecimento dedutivo deduz novas informações baseado na aplicação de regras lógicas predefinidas de dedução sobre dados existentes. A data mining apóia o conhecimento indutivo, que descobre novas regras e padrões nos dados fornecidos. O conhecimento pode ser representado de várias formas. Em um senso não estruturado, ele pode ser representado por regras ou por lógica proposicional. Em uma forma estruturada, ele pode ser representado por árvores de decisão, redes semânticas, redes neurais ou hierarquias de classes ou frames. Portanto as

próximas seções, serão úteis para descrever os modos e formas de descoberta de conhecimento durante a aplicação do data mining.

7.1.1. Regras de Associação

A descoberta de conhecimento por regras de associação é uma das principais tecnologias em data mining e possui maior número de aplicações práticas. Essas regras representam padrões de relacionamento entre itens de uma base de dados.

Uma das aplicações mais típicas para as regras de associação é a análise de transações de compras, um processo que examina padrões de compras de consumidores para determinar produtos que costumam ser adquiridos em conjunto. Portanto, utilizaremos este modelo para melhor compreender e exemplificar como são obtidas as regras de associação. Sendo assim, podemos imaginar um exemplo de regra de associação obtida a partir de uma base de dados que registra as compras de um supermercado, sendo: {Peça de picanha} $\Rightarrow$ {Cerveja}. Esta regra de associação indica que consumidores que compram o produto {Peça de picanha} estão mais propensos a adquirir também o produto {Cerveja}.

Entrando um pouco mais a fundo nas definições formais de regras de associação, podemos considerar um cenário onde temos $I = \{I_1, I_2, ...I_n\}$ um conjunto de itens, e D uma base de dados de transações, em que cada transação T é formada por um conjunto de itens onde $T \subseteq I$. Uma regra de associação é uma implicação da forma $A \Rightarrow B$, onde A e B podem ser conjuntos compostos por um ou mais itens, $A \subset I$, $B \subset I$, e necessariamente $A \cap B = \varnothing$. Neste caso, o conjunto A é chamado de antecedente da regra e o B é chamado de conseqüente.

Para que a regra de associação seja do interesse de um pesquisador ou analista, a regra precisa satisfazer algumas medidas. Duas medidas de interesse comum fornecem suporte e confiança.

O suporte para uma regra $A \Rightarrow B$, representa a porcentagem de transações da base de dados que contêm os itens de A e B, indicando a relevância da mesma. Se o suporte é baixo, isso indica que não existe nenhuma evidência significativa que os itens de A e B ocorram juntos, já que o conjunto de itens ocorre apenas em uma pequena fração das transações. Já a sua medida de confiança representa, dentre as

transações que possuem os itens de A, a porcentagem de transações que possuem também os itens de B, indicando a validade da regra. Para compreender melhor os conceitos abordados acima, tomaremos como exemplo uma pequena amostra de uma base de dados extraída do data warehouse, que armazena as compras efetuadas por clientes de um dado supermercado:

Id Transação	Itens Comprados
1012	lingüiça, alface, leite
1090	alface, cerveja, pão, leite, carne
1287	açucar, pão, carne, manteiga
1321	leite, vinagre, alface, suco
1479	pão, leite, açucar, chá
1503	café, pão, leite, iogute
1678	suco, vinagre, refrigerante, lingüiça

Figura 25 – Banco de dados amostral do supermercado.

Como podemos notar na figura, cada registro deste banco de dados armazena os itens adquiridos por um determinado cliente. Um exemplo de regra de associação que poderia ser minerada neste banco de dados seria a associação entre os produtos {pão} $\Rightarrow$ {leite}. É possível perceber que dentre as sete transações da amostra, três contêm os produtos {pão} e {leite}. Neste caso, encontramos uma medida de suporte de 3 ÷ 7 = 42,86%. Como podemos observar, encontramos três transações que possuem os produtos {pão} e {leite} juntos e quatro transações que possuem o produto {pão}. Assim, concluímos que a confiança da regra {pão} $\Rightarrow$ {leite} pode ser medida da seguinte maneira: 3 ÷ 4 = 75%. Com esses valores, podemos concluir que 75% dos consumidores que compram pão, também compram leite.

Porém, suporte e confiança não necessariamente são medidas proporcionais. O principal problema da aplicação do data mining para encontrar regras de associação consiste em encontrar todas as regras de associação que possuam suporte e confiança maiores ou iguais aos limites especificados pelo usuário. Para facilitar na resolução do problema, podemos fragmentá-lo em duas etapas de solução:

I. Gerar todos os conjuntos de itens que esteja acima dos limites de suporte estabelecidos.

II. Para cada conjunto de itens acima dos limites de suporte estabelecidos encontrados na etapa anterior, encontrar os que possuem um mínimo de confiança conforme especificado pelo usuário.

Segundo Navathe (2005), descobrir todos os conjuntos de itens que satisfaçam os limites de suporte especificados pelo usuário/analista junto com o valor para seu suporte é um problema significativo se a cardinalidade do conjunto de itens é muito alta. Um supermercado pode conter milhares de itens. O número de conjunto de itens distintos é 2^m , onde m é o número de itens, e o levantamento do suporte para todos os conjuntos de itens possíveis exige intenso esforço computacional. Porém, existem alguns algoritmos específicos que possuem propriedades capazes de reduzir o espaço de busca combinatória, como por exemplo o algoritmo Apriori.

Note que no processo de mineração de dados, as hipóteses e os padrões são automaticamente extraídos da base de dados pelas ferramentas, diferente de outras aplicações OLAP, onde o analista testa a sua hipótese e executa consultas contra a base de dados. Como dito anteriormente, através da descoberta de regras de associação torna-se possível, por exemplo, planejar a distribuição e a proximidade entre produtos nas prateleiras e combinar itens associados em qualquer campanha ou iniciativa de marketing ou vendas.

7.1.2. Classificação

Segundo Navathe (2005), a classificação é o processo de encontrar um modelo que descreva classes diferentes de dados, onde as classes são predeterminadas. Por exemplo, em uma aplicação bancária, clientes que possuem um cartão de crédito podem ser classificados como “risco baixo”, “risco médio” e “risco alto”. Assim, esse tipo de atividade é também chamado de aprendizado supervisionado. Dado que este modelo é construído uma vez, ele pode ser utilizado para classificar novos dados e atribuir, por exemplo, uma abordagem estratégica diferenciada para cada cliente dependendo do nível de risco a ele atribuído. O

primeiro passo do modelo de aprendizado é realizado usando um treinamento com um conjunto de dados que já foram classificados. Cada registro nos dados de treinamento com um conjunto de dados de treinamento contém um atributo, chamado rótulo de classe, que indica a que classe o atributo pertence. O modelo produzido normalmente está no formato de uma árvore de decisão ou um conjunto de regras.

Uma árvore de decisão é uma representação gráfica da descrição de cada classe ou, em outras palavras, uma representação das regras de classificação. Dado uma base de dados com diversos atributos e variáveis (colunas), a primeira etapa para montar uma árvore de decisão é definir um atributo alvo, ou seja, o indicador chave ou principal critério utilizado para tomada de decisão. Após a definição do atributo alvo, o próprio software irá sugerir um nó para partição das amostras através da variável que traga o maior critério para separação, ou seja, aquele que maximize a medida de ganho de informação. Assim, as regras ou nós serão subsequentemente classificados como nós dos nós anteriores, levando sempre em consideração o ganho de informação que a partição das amostras possa trazer. Desta forma, a regra mais importante e que mais discrimina clientes bons de clientes maus de acordo com o atributo alvo, será apresentada na árvore como primeiro nó, e as regras menos relevantes, serão mostradas nos nós subseqüentes. Para melhor compreensão, adotaremos o exemplo da figura 26.

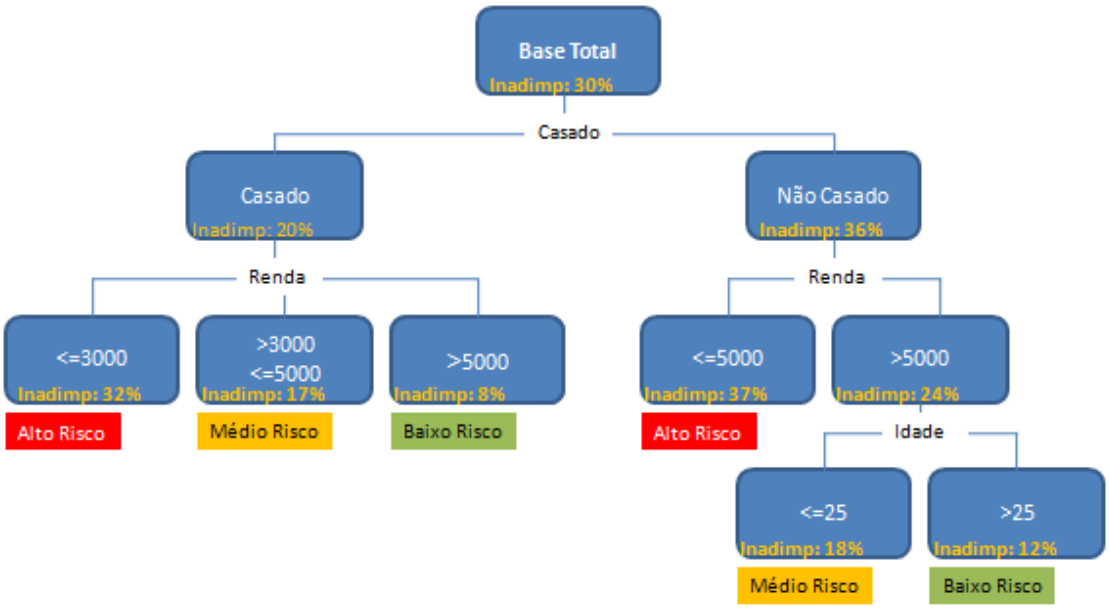

Figura 26 – Exemplo: árvore de decisão.

A ilustração acima demonstra a aplicação de uma árvore de decisão utilizada para segmentação da carteira de clientes de cartão de crédito de um banco. Como podemos observar, neste exemplo, o atributo alvo utilizado foi o fator inadimplência. Portanto, este é o que classifica um cliente dentro da carteira como sendo bom ou mau e é a partir dele que pretendemos criar regras através de outros atributos, que nos permitam classificar de antemão um cliente que tem propensão a ser inadimplente. Apenas para ilustrar melhor, o atributo inadimplência, neste caso, pode ser uma flag dentro da base de dados em questão que quando setado com o valor 1, indique um cliente inadimplente. Sendo assim, na base total temos um índice de 30% de inadimplência. O primeiro nó sugerido para partição da amostra é o nó de estado civil quebrado entre casado e não casado. Isso significa que dentre todas as outras variáveis presentes na base de dados em questão, o fator estado civil é o que mais discrimina clientes inadimplentes de clientes não inadimplentes. Em outras palavras, o estado civil é a variável que tem a maior capacidade de maximizar a medida de ganho de informação. Apenas para mencionar, o índice de inadimplência que inicialmente era de 30% cai para 20% quando analisamos apenas a população de casados. A partir daí, as amostras serão particionadas recursivamente sempre considerando os atributos que tragam o maior ganho possível e permita maior discriminação entre inadimplentes e clientes bons que pagam em dia. O intuito neste caso é efetuar a segmentação da carteira de acordo com diferentes níveis de risco. Assim, a instituição financeira poderá adotar diferentes estratégias dependendo do nível de risco do cliente. Como podemos observar, conseguimos segmentar a base e encontrar uma população de baixo risco que possui um índice de 8% a 12% de inadimplentes. Assim, para este perfil de cliente, seria coerente uma campanha para oferta de novos produtos e talvez um aumento do limite de crédito. É claro que existem outras variáveis de perda e rentabilidade a serem consideradas para esse tipo de tomada de decisão, mas a princípio, parece razoável. Assim como para os clientes com perfil semelhante ao encontrado pela segmentação de alto risco, seria razoável uma atenção maior na hora de ofertar, além de adotar uma estratégia de cobrança especial com intuito de mitigar perdas de crédito.

7.1.3. Agrupamento

Consiste em particionar ou segmentar uma população de eventos ou novos itens em conjuntos de elementos similares. O objetivo do agrupamento é colocar os registros em grupos, de tal forma que os registros de um grupo sejam similares aos demais do mesmo grupo e diferentes daqueles dos demais grupos. É importante destacar que, diferente da técnica de classificação, no agrupamento os dados são divididos sem uma amostra de treinamento, o que é chamado de aprendizado não-supervisionado.

O processo de clustering ou agrupamento pode ser aplicado, por exemplo, na área de marketing como uma ferramenta de auxílio na descoberta de grupos distintos de clientes, e uso deste conhecimento para criar campanhas dirigidas; em empresas de seguros, na identificação de grupos de assegurados com alto custo de sinistro; no estudo de terremotos, auxiliando na identificação de epicentros e seu agrupamento ao longo de falhas geológicas.

7.1.4. Padrões Sequenciais

Segundo Navathe (2005), a descoberta de padrões seqüenciais é baseada no conceito de uma sequência de conjuntos de itens. Consideramos que transações como as de um carrinho de supermercado que discutimos anteriormente sejam ordenadas por hora da compra.Essa ordenação gera uma sequência de conjuntos de itens. Por exemplo, {açúcar, pão, carne, manteiga}, {leite, vinagre, alface, suco}, {pão, leite, açúcar, chá} pode ser uma sequência de conjunto de itens com base em três visitas do mesmo cliente.O suporte para um sequência S de um conjunto de itens é a porcentagem do conjunto U de sequências das quais S é um subsequência. Neste exemplo, {açúcar, pão, carne, manteiga}, {leite, vinagre, alface, suco} e {leite, vinagre, alface, suco}, {pão, leite, açúcar, chá} são consideradas subseqüências. O problema de identificar padrões seqüenciais é, assim, encontrar todas as subsequências de um dado conjunto de sequências que tem um suporte mínimo definido pelo usuário. A sequência S_1, S_2, S_3, ... é um precursor do fato de

que um cliente que compra o conjunto de itens S1 esteja propenso a comprar S_2 e S_3, e assim por diante. Essa previsão é baseada na frequência (suporte) dessa sequência no passado. Vários algoritmos têm sido pesquisados para detecção de sequência.

7.1.5. Padrões em Série Temporais

Segundo Navathe (2005), séries temporais são sequências de eventos; cada evento pode ser um tipo fixo de uma transação. Por exemplo, o preço fechado de uma ação ou um fundo constitui uma série temporal. Para uma série temporal, podem-se estabelecer vários padrões analisando as sequências e subsequências, conforme feito anteriormente. Por exemplo, podemos encontrar o período durante o qual a ação caiu ou permaneceu estável por n dias, ou podemos encontrar o período mais longo durante o qual a ação teve uma flutuação de não mais que 1% sobre o preço previamente fechado, ou o trimestre durante o qual a ação teve o maior percentual de ganho ou percentual de perda. Séries temporais podem ser comparadas por meio do estabelecimento de medidas de similaridade para empresas cujas ações tenham um comportamento similar.

7.2. Aplicações de Data Mining

As técnicas de data mining podem ser aplicadas em grande variedade de contextos e áreas funcionais dentro das empresas. Em geral, as áreas com maior retorno de investimento esperado são:

- Finanças: aplicações incluem análise de crédito de clientes, segmentação de contas a receber, análise de performance de investimentos financeiros; avaliação de opções de financiamento; e detecção de fraudes.
- Produção: aplicações envolvem otimização de recursos como máquinas, força de trabalho, e materiais; projeto ótimo de processos de fabricação, layouts de chão de fábrica; e projeto de produto, como de automóveis baseados nos requisitos dos clientes.

- Marketing: aplicações como análise de comportamento do consumidor baseadas em padrões de consumo; definição de estratégias de marketing incluindo propaganda, localização de lojas e mala direta direcionada; segmentação de clientes, lojas ou produtos; projeto de catálogos layouts de lojas e campanhas de publicidade.
- Saúde: aplicações incluem descoberta de padrões em imagens radiológicas, análise de efeitos colaterais de remédios e efetividade de certos tratamentos; otimização de processos dentro de um hospital, relação de saúde do paciente com qualificações do médico.

CONCLUSÃO

Diante da competitividade acirrada entre as organizações inseridas no mercado contemporâneo, torna-se crucial a busca contínua por vantagens competitivas que tragam perenidade. Nos dias de hoje, a única certeza que as grandes empresas podem ter em relação aos seus respectivos futuros, é que se estas não tiverem capacidade suficiente de inovação e de se adequar de forma rápida ao dinamismo do mercado, se tornarão obsoletas e muito provavelmente encerrarão as suas atividades.

É buscando vantagens e diferenciais competitivos que as empresas investem em tecnologia da informação de forma alinhada com as estratégias e objetivos gerais dos negócios de forma a proporcionar um ambiente favorável ao desenvolvimento dinâmico dos negócios.

Dentre a gama infinita de tecnologias da informação disponíveis hoje no mercado, o Data Warehouse é adotado dentro das grandes empresas como uma ferramenta que fornece maior eficiência nos processos de tomada de decisão. A partir da integração dos dados disponíveis em ambientes operacionais e externos, esta tecnologia é capaz de prover suporte para que os administradores possam obter um conhecimento do negócio de forma mais ampla, maximizando o atendimento e o nível de satisfação dos clientes e otimizando de forma mais racional os processos de negócio e estratégias de acordo com o planejamento e as metas da empresa. Através de mecanismos e técnicas sofisticadas de Data Mining, utilizadas de forma a potencializar as funcionalidades do Data Warehouse, ainda é possível obter maior ganho de informações, além da descoberta de conhecimento e de novas formas de inovar e fazer negócio. Resumidamente, o uso do Data Warehouse combinado com as outras ferramentas utilizadas de forma complementar mencionadas durante o trabalho, permitem a obtenção de conhecimento a partir do uso correto dos dados disponíveis dentro das organizações, de forma a agregar valor ao negócio e proporcionar um ambiente composto por um número maior de variáveis concretas que possibilitam maior eficiência e precisão nos processos decisórios.

É importante frisar que, neste contexto, a tecnologia não substitui o fator humano, principalmente no que diz respeito aos processos de tomada de decisão. O

Data Warehouse apenas fornece informações que favorecem uma decisão mais segura, mas isso não descaracteriza a responsabilidade e a capacidade de interpretação de um gerente de alto escalão ou de quem quer que seja o tomador de decisão.

A implantação da tecnologia Data Warehouse demanda altos investimentos, e por isso é imprescindível que o projeto esteja alinhado com as metas e estratégias gerais da empresa e seja muito bem dimensionado de forma a atender as necessidades do negócio com performance aceitável, justificando o retorno do investimento.

Por fim, é importante destacar que através deste trabalho ainda foi possível conhecer diversas formas como o Data Warehouse pode ser empregado nas várias áreas dentro das organizações, sua origem, evolução, características técnicas e como a ferramenta contribui para agregar valor no negócio e proporcionar agilidade e precisão nos processos de tomada de decisão.

REFERÊNCIAS BIBLIOGRÁFICAS

GONÇALVES, Eduardo Corrêa : Data Mining de Regras de Associação. Disponível em: <http://www.devmedia.com.br/post-6533-Data-Mining-de-Regras-de-Associacao-Parte-1.html>. Acesso em 15 novembro de 2011.

HOKAMA, Daniele Del Bianco : A modelagem de dados no ambiente Data Warehouse. Disponível em: <http://meusite.mackenzie.com.br/rogerio/tgi/2004ModelagemDW.pdf>. Acesso em 15 outubro de 2011.

INMON, W. H. **Como Construir o Data Warehouse.** 2ª ed. Rio de Janeiro : Campus, 1997.

INMON, W.H. **Gerenciando Data Warehouse**. 1ª ed. Makron Books : São Paulo, 1999.

KIMBALL, Ralph. **Data Warehouse Toolkit: guia completo para modelagem dimensional.** 2ª ed. Campus : Rio de Janeiro, 2002.

NAVATHE, Shamkant B. **Sistemas de Banco de Dados**. 4ª ed. Pearson Addison Wesley : São Paulo, 2005

STAIR, Ralph M. **Principios de Sistemas de Informação**. 9ª ed. Editora Pioneira : São Paulo, 2010.