

Aula 4

Sistema Gerenciador de Banco de Dados (SGBD)

Profª Vivian Arianne Barausse de Moura

Conversa Inicial

- *Data warehouse*
- Conceitos básicos sobre *data warehouse*
- Modelagem de dados para *data warehouse*
- Projeto de *data warehouse*
- Ferramentas de *data warehouse (data mart)*
- Servidores Olap

Data warehouse (DW)

- O *data warehouse* oferece armazenamento, funcionalidades e responsividade para as consultas, além das capacidades dos bancos de dados orientados para a transação

- É projetado exatamente para dar suporte à extração, ao processamento e à apresentação eficiente para fins analíticos e de tomada de decisão
- Contém uma quantidade muito grande de dados de várias fontes

- Fontes que podem incluir bancos de dados de diferentes modelos de dados e às vezes arquivos adquiridos de sistemas e plataformas independentes



Definições das aplicações para uso do DW

OLAP – processamento analítico on-line	É um termo usado para descrever a análise de dados complexos do <i>data warehouse</i> . Nas mãos de trabalhadores do conhecimento habilitados, as ferramentas Olap utilizam capacidades de computação distribuída para análises que exigem mais armazenamento e poder de processamento do que pode estar localizada economicamente e eficientemente em um desktop individual.
DSS – sistemas de apoio à decisão	Também conhecidos como sistemas de informações executivas. Não confunda com sistemas de integração empresarial. Ajudam os principais tomadores de decisões de uma organização com dados de nível mais alto em decisões complexas e importantes.
Mineração de dados	É usada para descoberta do conhecimento, o processo de procurar novo conhecimento imprevisto nos dados. Estudaremos sobre este assunto na próxima aula.

Fonte: Adaptado de Elsmari, Navathe, 2011, p. 721.

Modelagem de dados para *data warehouse*

- Para modelagem de dados, de acordo com Elsmari e Navathe (2011), deve ser levada em consideração a tipologia
- Em modelos multidimensionais, ocorre o relacionamento dos dados em matrizes multidimensionais. Essas matrizes são chamadas de cubo de dados e podem ter mais de três dimensões, sendo chamadas de hipercubo

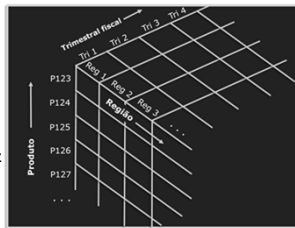
- Utilizando como exemplo uma planilha de vendas regionais por produto, para determinado período, os produtos poderiam ser mostrados como linhas, e as receitas de venda para cada região compreenderiam as colunas

- A figura abaixo mostra essa organização, que é uma planilha-padrão de matriz bidimensional

	Região			
	Reg 1	Reg 2	Reg 3	...
Produto	P123			
	P124			
	P125			
	P126			
	...			

Fonte: Elsmari, Navathe, 2011, p. 721.

- Ao acrescentar uma dimensão de tempo, como os trimestres fiscais de uma organização, produz-se uma matriz tridimensional, como na figura ao lado



Categorias de produtos	Região		
	Região 1	Região 2	Região 3
Produtos 1XX			
Produtos 2XX			
Produtos 3XX			
Produtos 4XX			

Fonte: Elsmari, Navathe, 2011, p. 724.

		Região 1				Região 2
		Sub_reg 1	Sub_reg 2	Sub_reg 3	Sub_reg 4	Sub_reg 1
Estilos P123	A					
	B					
	C					
	D					
Estilos P124	A					
	B					
	C					
	D					
Estilos P125	A					
	B					
	C					
	D					

Fonte: Elsmari, Navathe, 2011, p. 725.

Projeto de *data warehouse*

- Para a construção de um *data warehouse*, segundo Elsmari e Navathe (2011), os responsáveis devem ter uma visão ampla e antecipada de seu uso
- Não existe um meio de antecipar todas as consultas ou análises possíveis durante a fase do projeto

- O projeto deve aceitar especificamente a consulta ocasional, ou seja, acessar dados com qualquer combinação significativa de valores para os atributos nas tabelas de dimensão ou fatos

Etapas de aquisição de dados de um *data warehouse*

- 1 Os dados precisam ser extraídos de várias fontes heterogêneas, por exemplo, banco de dados ou outras entradas de dados, como aquelas que contêm dados do mercado financeiro ou dados ambientais.
- 2 Os dados precisam ser formatados por coerência dentro do *data warehouse*. Nomes, significados e domínios de dados de fontes não relacionadas precisam ser reconciliados.
- 3 Os dados precisam ser limpos para garantir a validade. A limpeza de dados é um processo complicado e complexo, que tem sido identificada como componente que mais exige trabalho na construção do *data warehouse*.
- 4 Os dados precisam ser ajustados ao modelo de dados do armazém. Baixar os dados de várias fontes. Devem ser instalados no modelo de dados no *data warehouse*.
- 5 Os dados precisam ser carregados no *data warehouse*. O grande volume de dados torna a carga dos dados uma tarefa significativa. São necessárias ferramentas de monitoramento para cargas, bem como métodos para recuperação de cargas incompletas ou incorretas.

Fonte: Adaptado de Elmasri e Navathe, 2011, p. 727.

Processos de armazenamento de dados

- ☐ Armazenamento de dados de acordo com o modelo de dados do armazém
- ☐ Criação e manutenção das estruturas de dados exigidas
- ☐ Criação e manutenção dos caminhos de acesso apropriados
- ☐ Fornecimento de dados variáveis no tempo à medida que novos dados são incluídos
- ☐ Suporte para atualização dos dados do *data warehouse*
- ☐ Atualização dos dados
- ☐ Eliminação dos dados

Fonte: Adaptado de Elmasri e Navathe, 2011, p. 727.

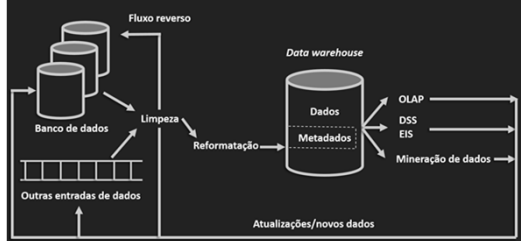
Elementos para projeto do *data warehouse*

- ☐ Projeção de uso
- ☐ O ajuste de modelo de dados
- ☐ Características das fontes disponíveis
- ☐ Projeto do componente de metadados
- ☐ Projeto de componente modular
- ☐ Projeto de facilidade de gerenciamento de mudança
- ☐ Considerações de arquitetura distribuída e paralela

Fonte: Adaptado de Elmasri e Navathe, 2011, p. 727.

Ferramentas de *data warehouse*

Exemplo de transações no modelo de cesta de mercado



Fonte: Karmakromani, 2008, p. 723.

Características diferenciadoras dos *data warehouses*

- ☐ Visão conceitual multidimensional
- ☐ Dimensionalidade genérica
- ☐ Dimensões e níveis de agregação ilimitados
- ☐ Operações e restrições entre dimensões
- ☐ Tratamento dinâmico de matriz esparsa
- ☐ Arquitetura cliente-servidor
- ☐ Suporte para múltiplos usuários
- ☐ Arquitetura cliente-servidor
- ☐ Acessibilidade
- ☐ Transparência
- ☐ Manipulação intuitiva de dados
- ☐ Desempenho de relatório consistente
- ☐ Recurso de relatório flexível

Fonte: Adaptado de Elmasri e Navathe, 2011, pag. 721.

Data warehouse

<i>Data warehouses</i> em nível empresarial	São imensos projetos que exigem investimento maciço de tempo e recursos.
<i>Data warehouses</i> virtuais	Oferecem visões de bancos de dados operacionais que são materializadas para acesso eficiente.
<i>Data marts</i>	Em geral, são voltados para um subconjunto da organização, como um departamento, e possuem um foco mais estreito.

Fonte: Adaptado de Elmasri, Navathe, 2011, p. 722.

Servidores Olap

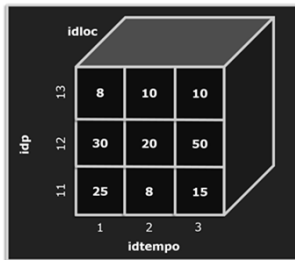
- De acordo com Ramakrishnan (2008, p. 704), "as aplicações de Olap são dominadas por consultas *ad hoc* complexas. Em termos de SQL, essas consultas envolvem operadores de agrupamento e agregação"

- A maneira natural de pensar sobre consultas Olap típicas é em termos de um modelo de dados multidimensional. Nele, o foco é e uma coleção de medidas numéricas, em que cada medida depende de um conjunto de dimensões

- Exemplo baseado em dados de vendas:
 - O atributo de medida do exemplo é vendas
 - As dimensões são produto, local e tempo

- Identificamos um produto por um identificador único *idp* e, analogamente, identificamos o local por *idloc* e o tempo por *idtempo*
- Consideramos as informações de vendas como organizadas em um *array* tridimensional Vendas, como mostra a figura a seguir

Vendas: um conjunto de dados multidimensional



Fonte: Ramakrishnan, 2008, p. 706.

- Os sistemas Olap realmente armazenam dados em um *array* multidimensional (implementado sem a suposição usual de linguagem de programação de que o *array* inteiro cabe na memória)
- Alguns dos atributos de uma dimensão descrevem a posição de um valor de dimensão com relação a essa hierarquia de valores de dimensão subjacente

- As hierarquias de produto, local e tempo de nosso exemplo aparecem no nível de atributo, na figura a seguir



Fonte: Ramakrishnan, 2008, p. 708.

- As informações sobre dimensões também podem ser representadas como uma coleção de relações

Locais (*idloc*: integer, *cidade*: string, *estado*: string, *país*: string)
 Produtos (*idp*: integer, *nomep*: string, *categoria*: string, *preço*: real)
 Tempos (*idtempo*: integer, *data*: string, *semana*: integer, *mês*: integer, *trimestre*: integer, *ano*: integer, *flag_feriado*: boolean)

Fonte: Ramakrishnan, 2008, p. 708.

Referências

- ELSMARI, R.; NAVATHE, S. B. Sistemas de banco de dados. 6. ed. São Paulo: Pearson Education do Brasil, 2011.
- RAMAKRISHNAN, R.; GEHRKE, J. Sistemas de gerenciamento de bancos de dados. 3. ed. Porto Alegre: McGraw Hill, 2008.

