

**Estácio de Sá – Juiz de Fora**

**Pós Graduação em Administração de Banco de  
Dados**

**Prof. Msc. Evaldo de Oliveira**

## Sobre o Módulo

- Conhecendo os alunos e o professor...
- Introdução
- Características de um SGBD
- Controle de Redundância
- Metadados
- Nível de Acesso
- Integridade
- O DBA (*Database Administrator*)
- Administração de Banco de Dados

# Introdução

# Introdução

- **Dados**
  - Dados são fatos que podem ser armazenados  
ex: nomes, telefones, endereços;
- **Informações:**
  - Informação é o resultado do processamento, manipulação e organização de dados de forma qualitativa ou quantitativa;

# Introdução

- **Banco de Dados (BD)**

- O que é um BD?

- Um banco de dados é uma coleção de dados relacionados, onde dados são fatos que podem ser gravados, usando ou não, um programa de computador (ELMASRI e NAVATHE, 2005);

- Propriedades de um BD:

- Representa aspectos do mundo real;
    - Projetado, construído e “povoado” com dados para um propósito específico;

## Introdução

- **Sistema Gerenciador de Banco de Dados (SGBD)**

- O que é um SGBD?

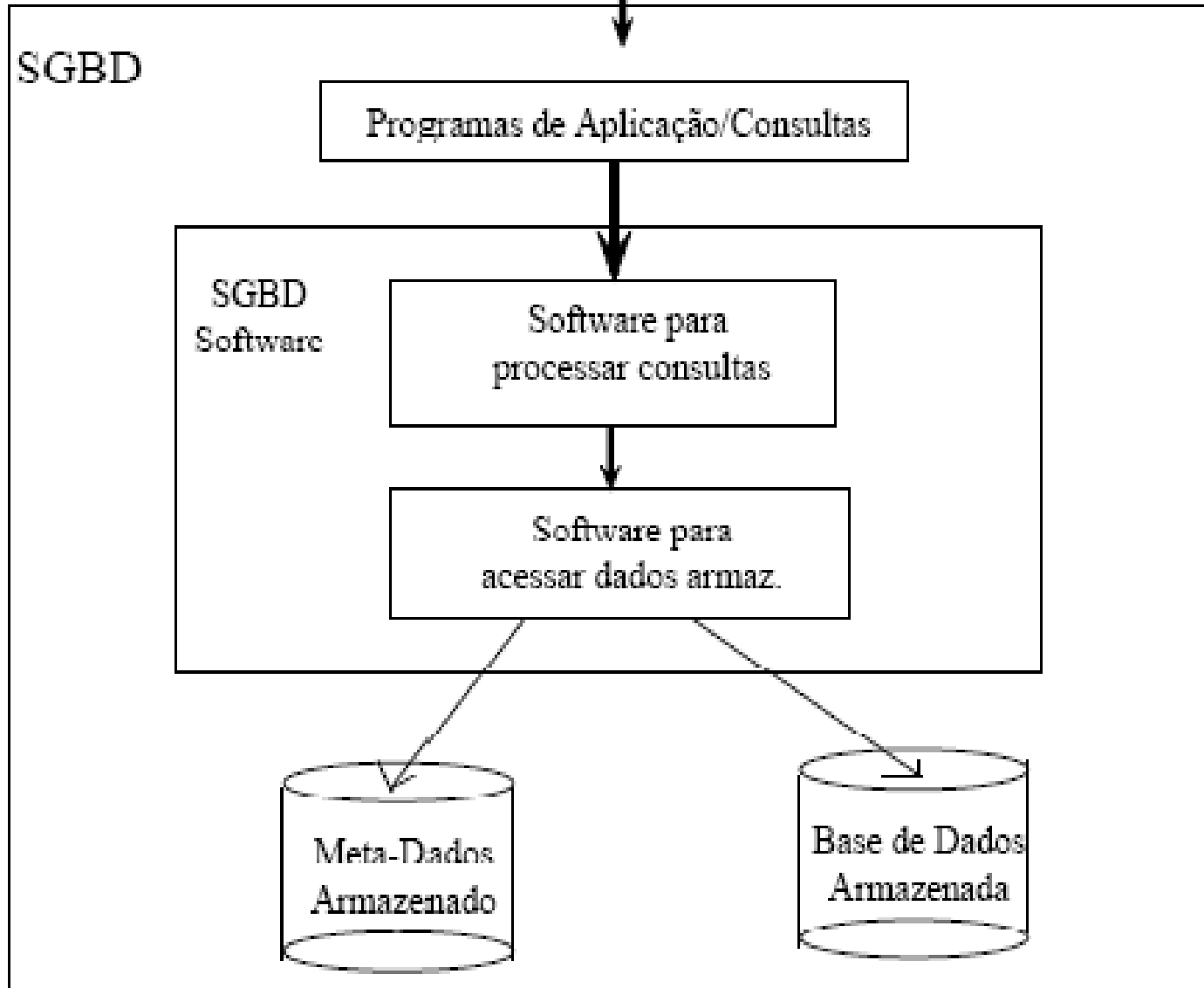
- Uma coleção de programas que permite aos usuários criar e manipular um banco de dados (ELMASRI e NAVATHE, 2005);

- Propósitos de um SGBD:

- Facilita processos de *definição, construção, manipulação e compartilhamento* de banco de dados entre vários usuários e aplicações;

# Introdução

Usuários



# Introdução

- **Quando usar um SGBD?**
  - Quando as funcionalidades de um SGBD são relevantes na manutenção e controle dos dados da aplicação:
    - Criação de Índices;
    - Controle do nível de acesso;
    - Utilitários de Backup e recuperação de dados;
    - etc...
  - Considerar vantagens adicionais:
    - Garante padronizações;
    - Possui Flexibilidade;
    - Reduz o tempo de desenvolvimento de uma aplicação;
    - Disponibiliza a informação atualizada;



# Introdução

- **Quando não usar um SGBD?**
  - Quando as funcionalidades não são necessárias;
    - Aplicação exige mais desempenho;
    - Investimento inicial alto, em geral: hardware extra;
  - Considerar situações não favoráveis:
    - A base de dados e as aplicações são simples, bem definidas e sem perspectivas de mudanças;
    - Requisitos de tempo real;
    - Não necessita acesso concorrente aos dados;

# Introdução

- **Propósitos de um SGBD**

- Definição:
  - Implica em especificar os tipos de dados, as estruturas e as restrições para os dados;
- Construção:
  - Visa armazenar os dados em alguma mídia;
- Manipulação:
  - Permite usar funções de pesquisa, atualização e recuperação de dados;
- Compartilhamento:
  - Permite múltiplos usuários e programas acessar o banco de dados de forma concorrente;

# Introdução

- **Funções importantes de um SGBD**
  - **Proteção:**
    - Inclui a proteção do sistema contra o mau funcionamento ou falhas (*crashes*) no hardware ou software;
  - **Segurança:**
    - Possui mecanismos que protegem os dados contra acessos não autorizados ou maliciosos;

# Introdução

- **Serviços de um SGBD**

- Redundância Controlada;
- Compartilhamento dos Dados;
- Concorrência;
- Reconstrução;
- Acesso controlado;
- Segurança;
- Restrições de Integridade;
- Armazenamento;
- Distribuição (DBLinks, Banco de Dados Distribuídos);

# Introdução

- **Redundância Controlada**

- A mesma relação de dados distribuída ou compartilhada por uma ou várias aplicações, visando otimizar consultas ou extração de dados. O mapeamento e a criação destas relações redundantes são mantidas pelo SGBD e documentadas pelos ADs e ABDs (DBAs);

- Ex: Sistemas distribuídos - um mesmo dado é armazenado em vários computadores, permitindo acesso rápido a partir de qualquer um deles

# Introdução

- **Redundância não Controlada**

- Acontece quando a responsabilidade pela manutenção da sincronia entre as diversas relações está sob responsabilidade do usuário.
- Redundância não Controlada leva a :
  - Redigitação de Dados;
  - Inconsistência dos Dados: os dados podem não representar corretamente a realidade. Imagine que o usuário alterou o preço de um produto no sistema de compra mas não alterou no sistema de vendas;
  - Dificuldade na extração de informações: os dados projetados para atender uma aplicação específica podem gerar dificuldade para o cruzamento dos dados;

# Introdução

- **Características do uso de Banco de Dados**
  - Natureza autodescritiva;
  - Isolamento entre programas e os dados e a abstração dos dados;
  - Suporte para múltiplas visões dos dados;
  - Compartilhamento de dados e processamento de transações de multiusuários;

# Introdução

- **Natureza Autodescritiva**
  - Definição ou descrição da estrutura do banco de dados (*metadados*);
- **Isolamento entre programas e Dados**
  - Permite independência programa-dados;
- **Suporte para as Múltiplas visões dos Dados**
  - Permite aos usuários a elaboração de consultas permitindo o resultado de um subconjunto de um banco de dados;



# Introdução

- **Compartilhamento de Dados e o Processamento de Transação com Multiusuários**
  - Um SGBD deve permitir o controle de concorrência para que muitos usuários possam atualizar um mesmo dado de forma segura;
  - Por exemplo, em um *Sistema de Controle de Trem*, o SGBD deve garantir que cada locomotiva possa ser anexada ou desanexada somente por um agente de cada vez, para registrar uma composição de trem;
  - Aplicações deste tipo permitem o processamento de transações on-line;
  - Uma regra do SGBD multiusuário é garantir que as transações concorrentes operem corretamente;

# Introdução

- **Compartilhamento de Dados e o Processamento de Transação com Multiusuários (continuação...)**
  - O conceito de **transação** tornou-se fundamental para muitas aplicações de banco de dados;
  - Uma transação é uma rotina em execução ou processo que inclui um ou mais acessos ao banco de dados;
  - Cada transação deve executar um acesso logicamente correto ao banco de dados, se executado sem a interferência de outras transações;
  - O SGBD deve garantir algumas propriedades da transação:  
**isolamento**  
**e atomicidade;**

# Introdução

- **Isolamento**

- Garante que cada transação possa ser efetuada de forma isolada de outras transações;

- **Atomicidade**

- Garante que todas as operações em um banco de dados, em uma transação, sejam executadas ou não;

# Introdução

- **Restrição de Acesso**

- Quando vários usuários tem acesso à base de dados, em geral eles tem privilégios diferentes quanto à manipulação dos dados;
- Dados financeiros costumam ter acesso restrito e o SGBD deve ter um mecanismo de controle de segurança;
- O SGBD deve garantir que usuários autorizados realizam operações corretas na base de dados:
  - Identificação de usuários e de dados;
  - Autenticação de usuários;
  - Manutenção da matriz de autorização;

# Introdução

- **Restrição de Integridade**

- **Restrições inerentes ao modelo:** São observadas durante o levantamento de requisitos do sistema, e representam regras abstraídas do mundo real;
- **Restrições baseadas em esquema (ou restrições implícitas):** Restrições que podem ser declaradas de forma não procedural na DDL (chave primária, integridade referencial, restrição de domínio, etc.);
- **Restrições baseadas em aplicação:** Nesta categoria, enquadram-se todas as restrições que devem ser explicitamente especificadas no SGBD em questão, através de programação, ou por meio de instruções especiais (ELMASRI e NAVATHE, 2005)

# Introdução

- **Restrição de Integridade**

- **Manutenção da consistência da base de dados através da validação de restrições definidas sobre os dados:**
  - Controle de valores que um campo pode ter;
  - Por ex.: O número de vagões em um trem de minério pode variar entre 100 e 132.
- **Controle de relacionamento entre dois registros.**
  - Por ex.: Um vagão só pode pertencer a um trem formado;
- **O SGBD deve oferecer dois mecanismos:**
  - Especificação de restrições de integridade
  - Validação das restrições de integridade

# Introdução

- **Armazenamento Persistente**

- Armazenamento de objetos e estruturas de dados, desenvolvidas por meio das linguagens C++, Java ou tipos de registros criados em PASCAL;
- Programas necessitam converter estruturas complexas para formata compatível;
- Sistemas de Banco de Dados Orientado a Objetos são compatíveis com estruturas criadas em aplicações desenvolvidas em Java, C++, etc..
- Os sistemas tradicionais de banco de dados geralmente possuem problema de **separação por impedância**, ou seja, as estruturas de dados fornecidas pelos SGBDs são incompatíveis com as estruturas desenvolvidas em aplicações orientadas a objetos;

# Introdução

- **Armazenamento de Estruturas Para Processamento Eficiente**
  - Os SGBDs devem fornecer funcionalidades para a execução de *atualizações e consultas eficientemente*;
  - Estruturas especializadas para aumentar velocidade de pesquisa em disco:
    - Índices
    - Armazenamento temporário;
    - Processamento de consulta e otimização;
  - Porém, deve-se tomar alguns cuidados...



# Introdução

- **Garantia de Backup e Restauração**

- SGBD deve fornecer utilitários para backup, cópia e restauração:

- COPY;
    - RMAN (Restaure Manager. Utilitário do Oracle 10g);
    - BCP (Utilitário do SQLServer para exportar e importar tabelas e queries)
    - IMP;
    - EXP;
    - Restore;

# Introdução

## • Representação de Relacionamentos Complexos

- Relacionamentos usando subconsultas;
- Entre objetos complexos. Por exemplo, em consulta espacial: Localizar o ponto de ônibus mais próximo da Praça Sete;
- Relacionamentos Com  $n$  tabelas e views;
- **Alter Join**. Por exemplo, consultar todos os clientes que tenham ou não pedidos, mostrando também o número de pedido do cliente. Por exemplo:

```
select cli.nm_cliente, ped.cd_pedido  
from cliente cli  
      , pedido ped  
where cli.cd_cliente = ped.cd_cliente(+)  
order by cli.cd_cliente
```



Cliente	Pedido
Zebedeu	123650
Baraki Obama	985600
Pedro das Candongas	
George M. Bush	456213
Maysa Monjardim	

# Introdução

- **Implicações adicionais do Uso da Abordagem de um Banco de Dados**

- Potencial para garantir padrões:
  - Padrões para formatos e criação de atributos;
  - Terminologias (Dicionário de Dados);
  - Padronização do nível de acesso;
- Redução no Tempo de Desenvolvimento de Aplicações;
- Flexibilidade;
- Disponibilidade para atualizar informações para um grupo de usuários ou todos usuários;
- Economia de Escala:
  - Consolidação e compartilhamento dos dados;
  - O investimento em hardware e processadores , reduz o custo total da operação e gerenciamento do SGBD;

# Introdução

- **Aplicações Orientadas a Objetos e a Necessidade de Banco de Dados Mais Complexos**
  - Criação de Tipos;
  - Tabelas Aninhadas;
  - Herança de Tipos;
  - Armazenamento de textos, imagens (Objetos Longos ou BLOBs);
  - Representação de atributos multivalorados (VARRAY, LINESTRING, etc);
  - Exemplo: Sistemas de Informações Geográficas;

# Sistemas de Banco de Dados

# Sistemas de Banco de Dados

- **Linguagens de SGBD**
  - DDL (Data Definition Language)
  - DML(Data Manipulation Language)
  - DCL(Data Control Language)

# Sistemas de Banco de Dados

- **Utilitários do Sistema de Banco de Dados**
  - Carregamento (*loading*): usado para carregar os arquivos de dados existentes, por exemplo, arquivos texto;
  - Backup;
  - Reorganização de arquivos. Exemplo: Oracle COALESCE e SQLServer DBCC INDEXDEFRAG
  - Monitoramento de desempenho;

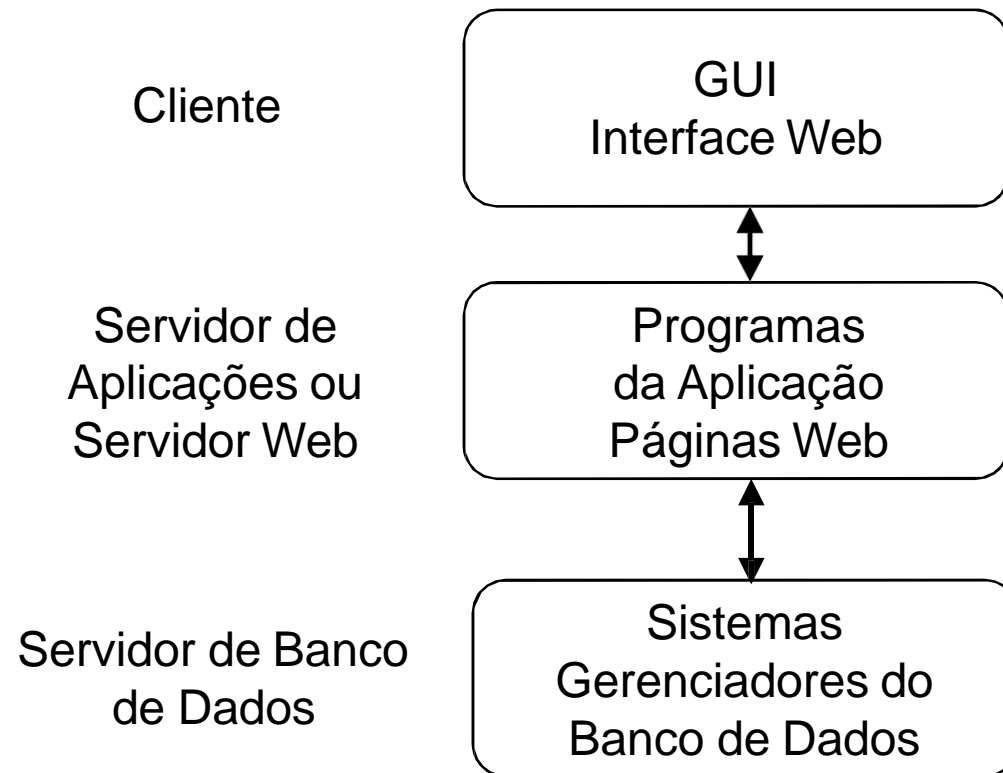
## Sistemas de Banco de Dados

- **Arquiteturas Cliente/Servidor de Duas Camadas**
  - Consultas elaboradas no *client* e no processadas no *server*;
  - Conexão via *client*;
  - Comunicação com o *server* via ODBC (*Open Database Connectivity*);
  - String de conexão elaborada pelo *client* da aplicação;



# Sistemas de Banco de Dados

- **Arquiteturas Cliente/Servidor de Três Camadas**



# Sistemas de Banco de Dados

- **Objetos do Banco de Dados**

- Tabelas
- Visões
- Índices
- Sequences
- Stored Procedures
- Functions
- Triggers
- Sinônimos

# **A área de Banco de Dados nas Empresas**

## **A área de Banco de Dados nas Empresas**

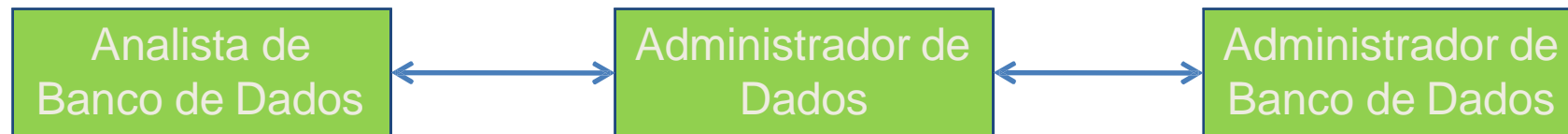
- Profissionais
- Padrões para Banco de Dados
- Vocon (Vocabulário Controlado)
- Processos da área de Tecnologia da Informação
- Certificações

# A área de Banco de Dados nas Empresas

- **Profissionais**

- Administradores de Banco de Dados (ABD ou DBA)
  - Monitora as Rotinas de Banco de Dados
  - Gerencia Usuários
- Administradores de Dados (AD)
  - Responsável pelo Padrão para Banco de Dados
  - Valida esquemas conceituais e físicos
  - Mantém o vocabulário controlado
  - Mantém os esquemas criados em ambiente de desenv.
- Analista de Banco de Dados (ou Analista de Sistemas)
  - Projetam os esquemas de banco de dados
  - Desenvolvem as rotinas de banco de dados

# A área de Banco de Dados nas Empresas



1. Faz a análise de Requisitos e elabora o esquema conceitual com base no padrão para banco de dados

1. Discute as regras de negócios com o Analista de Banco de Dados, valida o esquema conceitual com base no padrão para banco de dados e gera os *scripts* de banco em desenvolvimento

2. Ao finalizar a validação do esquema conceitual e o desenvolvimento da aplicação, os *scripts* de banco são enviados para produção

1. Antes de criar os *scripts* em produção, faz a análise do *storage* em banco de dados, e também a necessidade de criação de índices

# A área de Banco de Dados nas Empresas

- **Padrões para Banco de Dados**
  - Normalmente é desenvolvido e mantido pelo AD
  - Especifica as nomenclaturas e abreviaturas usadas para nomear os objetos de um banco de dados
  - Padroniza a geração de *scripts* para criação e/ou manutenção de um banco de dados
  - Serve de guia para os Analistas de Banco de Dados projetar e/ou desenvolver um banco de dados
  - Deve ser seguido por toda equipe de TI

# A área de Banco de Dados nas Empresas

- **VOCON (Vocabulário Controlado)**
  - É normalmente elaborado pelo AD
  - Pode ser uma base de dados que possui todas as nomenclaturas e abreviaturas criadas para nomear tabelas, atributos e diversos objetos do banco de dados
  - Nomenclatura para objetos de banco de dados
    - **cd** . *código*. cd\_produto. Código do Produto
    - **nm**. *nome*. Nm\_usuario. Nome do usuário
    - **fk** . *Foreign Key*. fk\_produto\_pedido



# **A área de Banco de Dados nas Empresas**

Exemplo de um Padrão para Banco de Dados

# A área de Banco de Dados nas Empresas

- **Montagem dos Scripts de Banco**

- 01\_create\_user.sql
- 02\_create\_role.sql
- 03\_create\_table.sql
- 04\_comments.sql
- 05\_alter\_table\_pk.sql
- 06\_alter\_table\_fk.sql
- 07\_create\_sequence.sql
- 08\_create\_synonyms.sql
- 09\_grants.sql

# O DBA

## *(Database Administrator)*

## O DBA

- Normalmente é bem remunerado
- É responsável por acompanhar e tomar medidas corretivas necessárias nos momentos adequados
- É imprescindível na “vida” de um banco de dados
- Figura humana peculiar, sempre deve estar atento e preparado para o “pior”
- Deve estar preparado e disponível 24x7

# O DBA

- Cabe ao DBA instalar o SGBD
- O DBA deve configurar o SGBD de forma a aproveitar e otimizar os recursos de hardware
- O DBA deve disponibilizar o SGBD para que possa ser utilizado de maneira segura pelos usuários
- O DBA deve conhecer a organização interna do SGBD tanto em nível físico (memória, processos, arquivos) quanto lógico (distribuição dos dados)
- O DBA deve ser responsável pela integridade dos dados, e em caso de quaisquer perdas (falha em disco, destruição acidental, etc.), oferecendo uma cópia que restaure o estado dos dados antes da perda

# O DBA

- Em resumo, o DBA deve:
  - Instalar o SGBD
  - Criar a base de dados, tabelas, índices e manter outros objetos do banco
  - Monitorar e otimizar o desempenho do SGBD
  - Manter o *storage* (capacidade de armazenamento)
  - Disponibilizar as bases de dados
  - Gerenciar e manter os usuários
  - Fazer cópias de segurança
  - Recuperar a base de dados
  - Ser pró-ativo e reunir-se periodicamente com Analistas de Banco de Dados e Desenvolvedores para obter as configurações ótimas para o Banco

# Administração de Banco de Dados na prática

# Administração de Banco de Dados na Prática

- Para administrar um SGBD é necessário dominar comandos SQL, como DDL, DML e DCL
- Para manter o SGBD deve-se utilizar aplicativos para acesso às bases de dados
- Deve-se conhecer os principais objetos
- Além disso, é importante saber o que é uma transação, quando começa e quando termina



# Administração de Banco de Dados na Prática

- **Na prática, o DBA deve compreender algumas características importantes sobre o SGBD**
  - Organização Física
  - Organização Lógica
  - Criação de Tabelas
  - Criação de Índices
  - Segurança e nível de acesso
  - *Backup e Recover*
  - Link entre bancos de dados

# Administração de Banco de Dados na Prática

- **Organização Física**

- É constituída na maioria das vezes por arquivos importantes para o funcionamento do SGBD
- Possui processos imprescindíveis para manutenção do SGBD
- Pode sofrer manutenção ou ser consultados por meio de comandos DCL ou DML

# Administração de Banco de Dados na Prática

- **Organização Lógica**

- Em muitos SGBDs uma *Tablespace* é o recurso que permite a organização lógica do banco
- Muitas vezes é necessário a criação de várias *Tablespaces*
- É possível ajustar a *Tablespace* por meio de comandos DDL

# Administração de Banco de Dados na Prática

- **Criação de Tabelas**

- Em alguns SGBDs a criação de tabelas pode ser segmentada
- É possível criar, alterar ou eliminar estes segmentos por meio de comandos DDL

# Administração de Banco de Dados na Prática

- **Criação de Índices**

- Em alguns SGBDs a criação de índices pode ser segmentada
- É possível criar, alterar ou eliminar estes segmentos por meio de comandos DDL

# Administração de Banco de Dados na Prática

- **Segurança e Nível de acesso**

- Permite a identificação do papel de um usuário no Banco de Dados
- Pode-se definir e configurar *schemas* e *privilégios*
- É possível alteração e eliminação de usuários por meio de comandos DCL
- Atribuição de privilégios de sistema, grupos e objetos aos usuários criados

# Administração de Banco de Dados na Prática

- ***Backup e Recover***

- A atividade de *Backup* é imprescindível
- Os SGBDs oferecem utilitários para cópias de segurança
- Podem ser feitas cópias lógicas e físicas de um banco de dados
- Os SGBDs também oferecem utilitários para recuperação de cópias de segurança

# Banco de Dados Oracle



## Oracle – Organização Física

- **A organização física do SGBD Oracle é estruturada por quatro tipos de arquivos**
  - Data Files
  - Control Files
  - Redo Log Files
  - Arquivo de Parâmetros

# Oracle – Organização Física

- **Data Files**
  - Contém dados ou metadados que constam em tabelas, índices, áreas temporárias, etc...
- **Control Files**
  - Armazenam a estrutura do banco de dados responsável, por exemplo, pela inicialização do banco de dados
- **Redo Log Files**
  - Mantém o histórico das transações efetuadas, que são úteis em caso de perda de dados

# Oracle – Organização Física

- **Data Files**
  - v\$dbfile ou v\$datafile (mais detalhado)
- **Control Files**
  - v\$controlfile
- **Redo Log Files**
  - v\$logfile

# Oracle – Organização Física

- **Processos**
  - Um processo representa um pequeno programa que executa tarefas ou instruções bem específicas, como conexão ao Servidor ou execução de uma stored procedure, etc...
- **Processo *Usuário***
  - Ocorre por meio de conexão
- **Processo *Servidor***
  - É iniciado ao receber requisições de processos *usuários*

# Oracle – Organização Física

- **Instância**

- União de estruturas de memória e processos
- Uma vez criada, pode-se iniciar a comunicação
- Instâncias no Oracle podem obtidas a partir da visão `v$instance`

## Oracle – Organização Física

- **Etapas de inicialização de uma instância**
  - Nomouting: Estado que permite a criação de banco de dados ou recuperação danos, como por exemplo, nos arquivos de controle
  - Mounting: Os arquivos de controle são lidos e determina-se a localização de demais arquivos
  - Opening: Os arquivos de dados e logs são abertos

## Oracle – Organização Física

- **O que ocorre durante um SELECT**
  - Requisição : *Cliente* para *Servidor*
  - Alocação de Memória : Reserva de espaço
  - Parse : Verificação sintática
  - Execução : aplica-se o Plano de Execução
  - *Fetch* : cria-se um conjunto de linhas, e retorno para o cliente

# Oracle – Organização Física

- **O que ocorre durante um UPDATE**
  - Requisição : *Cliente* para *Servidor*
  - Alocação de Memória : Reserva de espaço
  - Parse : Verificação sintática
  - Leitura : Verifica se há alocação de memória no Buffer, caso contrário realiza leitura física
  - Reserva : Reserva um bloco para caso a transação corrente seja desfeita
  - Bloqueio: As linhas que serão alteradas sofrem bloqueio
  - Log para valores antigos: Valores que serão alterados são gravados no Log Buffer
  - Atualiza: As linhas são modificadas
  - Notificação : Servidor informa ao Cliente que as alterações foram realizadas com sucesso



# Oracle – Organização Lógica

- ***Tablespaces***

- Uma *tablespace* representa uma entidade lógica capaz de reunir dados (de controle, temporários ou dados propriamente ditos) que guardem semelhanças entre si;
- É possível criar *tablespaces* para tabelas ou índices para ajudar na organização de dados com relação ao tamanho;
- Onde tabelas ou índices maiores podem ser organizados em *tablespaces* maiores, e vice-versa.

# Oracle – Organização Lógica

- **Parâmetros para Criação de Tablespace**
  - DataFile
  - Default Storage
  - OnLine
  - OffLine
  - AutoExtend

## Oracle – Organização Lógica

- Os valores de INITIAL, NEXT, etc. devem ser obtidos após análise dos *scripts* de criação das tabelas;
- A partir dos tamanhos de cada registro e previsão de volume de dados, pode-se especificar o tamanho máximo que uma tabela com seus índices alcançará;
- Recomenda-se superestimar esses valores

## Oracle – Organização Lógica

**Supondo que cada bloco tenha 2.048 bytes**

INITIAL        50        102,4 bytes  
NEXT           25        51,2 bytes  
PCTINCREASE    100

EXTENTS	TAMANHO	EXPANDE	NOVO	EM BLOCOS
1	100	0	100	50
2	100	50	150	75
3	150	100	250	125
4	250	200	450	225
5	450	400	850	425
6	850	800	1650	825

# Oracle – Organização Lógica

- ***Tablespaces***
  - Alteração
  - Eliminação

## Oracle – Tabelas

- **Criação Segmentada**
  - Uso de *TableSpace*
  - Parâmetro de Armazenamento

## Oracle – Tabelas

- **Alteração da Estrutura de Tabelas**
  - Adicionar Colunas
  - Renomear Colunas
  - Apagar Colunas
  - Adicionar PKs
  - Adicionar FKs
  - Apagar *Constraints*
  - Modificar Colunas
  - Desabilitar FK
  - Apagar Tabelas
  - Listar Estruturas

## Oracle – Índices

- **Criação Segmentada**
  - Uso de *TableSpace*
  - Parâmetro de Armazenamento
- **Deleção de Índices**



# Oracle – Restrição de Integridade

- **Restrição Baseada em Esquema**
  - Restrição de Chave:
    - Primary Key
    - Foreign Key
    - Check Constraint

## Oracle – Restrição de Integridade

- **Restrição Baseada em Aplicação**
  - Assertion
  - Triggers ou Gatilhos:
    - *Triggers* com Insert
    - *Triggers* com Update
    - *Triggers* com Delete

# Oracle – Segurança e Nível de Acesso

- **Usuários**
  - Criação
  - Privilégios de Sistema
  - Privilégio de Objeto
  - Criação de um usuário dono do *schema*

# Oracle – Segurança e Nível de Acesso

- **Roles**
  - Criação
  - Atribuição
  - Política de Segurança

# Oracle – Backup e Cópias de Segurança

- **Exportação**
- **Importação**
- **RMAN (*Recovery Manager*)**
  - Comandos para Backup e Restore
- **Comando COPY**

# Oracle – Banco de Dados Distribuídos

- Database Link
- *Heterogeneous Services*
- Criação de Links
- Consulta à Tabelas Distribuídas

**Estácio de Sá – Juiz de Fora**

**Pós Graduação em Administração de Banco de  
Dados**

**Prof. Msc. Evaldo de Oliveira**