

ADMINISTRAÇÃO DE BANCO DE DADOS

Prof. Celso M. furtado

Introdução

- Desde os tempos mais remotos da tecnologia da informação os dados tem sido o coração de tudo que os computadores fazem;
- Usuários entram dados e o computador processa, analisa, faz cálculos e então a saída ocorre;
- Notou o que há de comum? Tudo está relacionado com dados!
- À medida que a tecnologia se torna cada vez mais parte da vida cotidiana mais os dados fazem parte da nossa vida.

Introdução

- **Os dados vem de várias formas e formatos. Consideremos alguns exemplos:**
 - Pedidos que você faz na sua loja online preferida;
 - Problemas que são armazenados por uma pessoa em um call center;
 - Dados financeiros incluindo calculo de juros, taxas e investimentos gerados por bancos e empresas de investimentos;
 - Imagens geradas por equipamentos médicos complexos;
 - Inventário de produtos que são rastreados através de suas etiquetas RFID e informações sobre os pontos de venda;
 - Arquivos de imagem, vídeo e áudio carregados nas redes sociais;
 - Sites de compartilhamento de vídeo...

Introdução

- A quantidade de dados gerados estão crescendo a uma taxa explosiva;
- Pense na quantidade de vendas online ocorrendo todos os dias;
- Acrescente a isso as postagens efetuadas nas mídias sociais a partir de smartphones;
- Cada venda no varejo está amarrado a sistemas de rastreamento e inventário automatizados;
- **Praticamente tudo que fazemos e que envolve interação, comunicação ou comércio GERA DADOS!!**

Introdução

- O tamanho e quantidade de dados criados também estão crescendo conforme a tecnologia as envolve;
- Arquivos de imagem, vídeos e som tomam cada vez mais espaço nos discos dos computadores do que simples arquivos de texto;
- Dispositivos médicos e científicos como as imagens por satélite e imagens por ressonância magnética são exemplos de como dados grandes são gerados rapidamente e são parte da vida das pessoas.

Definindo um banco de dados

- Todos estes dados precisam residir em algum lugar, mais especificamente em um banco de dados;
- Banco de dados são programas complexos que catalogam e provem acesso aos dados;
- Embora os dados sejam armazenados em sistemas de armazenamento de disco, os programas de banco de dados gerenciam como os dados são armazenados e acessados.

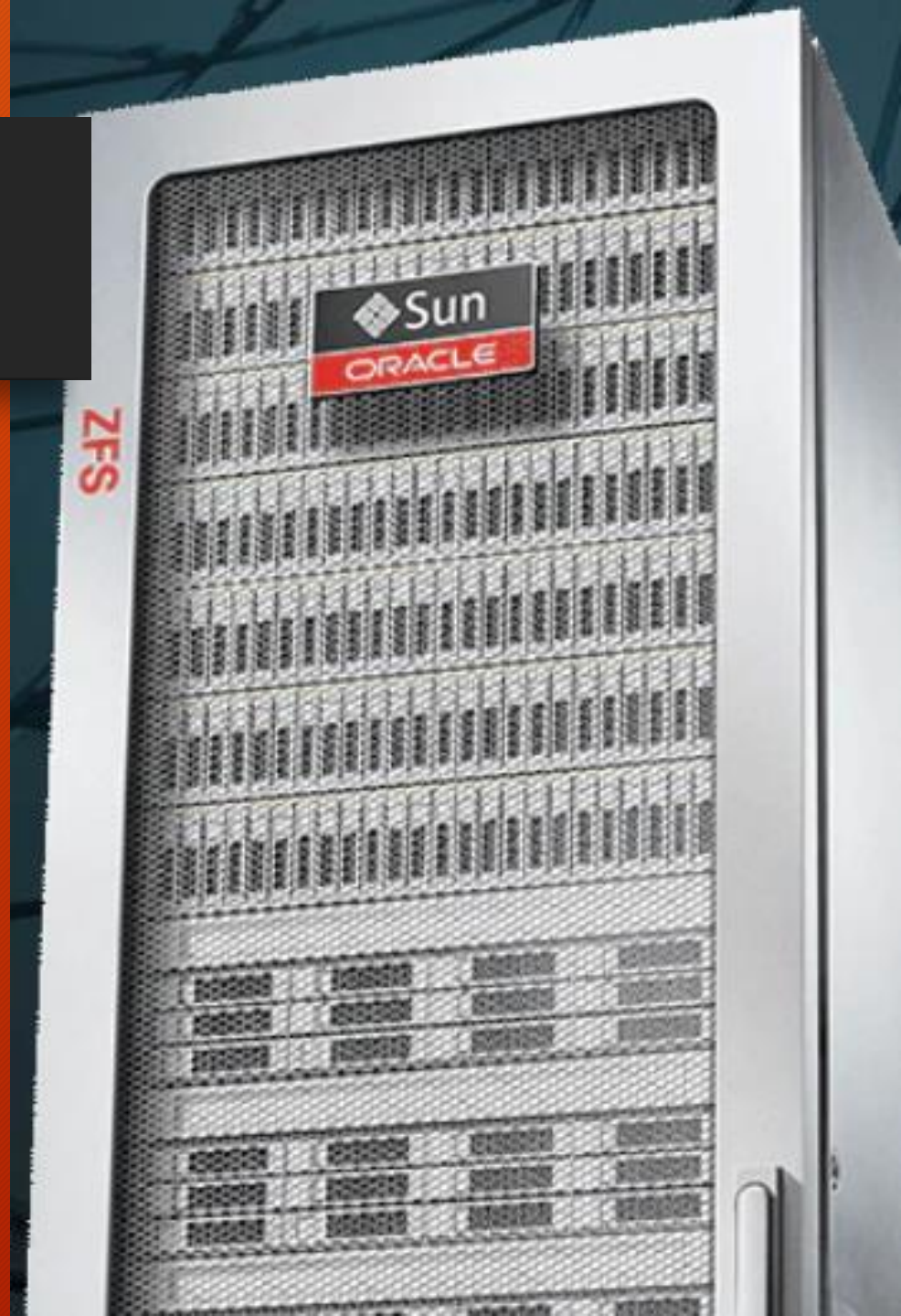
Definindo um banco de dados

- As empresas utilizam SGBD (Sistema Gerenciador de Banco de Dados) devido ao fato de que os seus dados muitas vezes **excedem** em **tamanho** e **complexidade** as formas de armazenamento mais comuns - planilhas, arquivos texto, etc.;
- A maioria das atividades empresariais e científicas envolvem tecnologias que requerem a criação, acesso, atualização ou exclusão dos dados em algum nível;
- Os SGBDs são os softwares que garantem que estas ações ocorram de forma rápida e eficiente.

Funções primárias de um SGBD

ARMAZENAMENTO

- Os dados precisam ser mantidos em um sistema computacional, de modo que estejam disponíveis quando necessário;
- As tecnologias de armazenamento podem variar, mas precisam ser rápidas e com grande capacidade. Muitas vezes são caros e complexos.



Funções primárias de um SGBD

ORGANIZAÇÃO

- Os dados devem ser armazenados de maneira lógica no disco de modo que possa ser acessado de forma rápida e eficiente.



Funções primárias de um SGBD

ACESSO

- Os dados devem ser acessados de forma rápida e eficiente;
- retornar os dados que o requisitante precisa é uma função chave.

HR_EMP_DETAILS_VIEW.rdb*

Layout Debug Tools Window Help

Connection Sadbox

100%

90%

Salary Rates by Regions

SALARY 2100

REGION NAME Americas

JOB TITLE	CITY	STATE PROVINCE	COUNTRY NAME	DEP
Stock Clerk	South San Francisco	California	United States of America	Shipp

SALARY 2200

REGION NAME Americas

JOB TITLE	CITY	STATE PROVINCE	COUNTRY NAME	DEP
Stock Clerk	South San Francisco	California	United States of America	Shipp
Stock Clerk	South San Francisco	California	United States of America	Shipp

SALARY 2400

REGION NAME Americas

JOB TITLE	CITY	STATE PROVINCE	COUNTRY NAME	DEP
Stock Clerk	South San Francisco	California	United States of America	Shipp
Stock Clerk	South San Francisco	California	United States of America	Shipp

Designer Preview HTML View Scripts Page 1 of 13

Funções primárias de um SGBD

SEGURANÇA

- Um SGBD determina **quem pode** acessar o banco de dados e **o que pode** ser feito com os dados;
- A segurança deve ser reforçada.



Funções primárias de um SGBD

ADICIONAR, ATUALIZAR E EXCLUIR

- Depois que os dados são adicionados ao banco ele **pode ser modificado e excluído**;
- O SGBD **gerencia as regras complexas** de como os dados são manipulados.



Funções primárias de um SGBD

ARMAZENAMENTO SEGURO

- Os SGBDs devem manter os dados **seguros e disponíveis** quando ocorrer alguma falha, como um dano a disco, por exemplo;
- **Backup** e **Recovery** são os processos utilizados para garantir que os dados não sejam **perdidos permanentemente**.



ORACL

Funções primárias de um SGBD

PROCESSO GERAL

Você abre o banco de dados e insere seus dados



O banco decide qual a **melhor** forma de catalogar os dados de maneira segura



Quando você quer acessar, modificar ou excluir os dados o SGBD checa suas configurações de segurança e implementa as requisições da maneira mais rápida e eficiente possível

Definindo um banco de dados

- Sistemas Gerenciadores de Banco de dados se parecem mais com um Sistema operacional do que uma simples aplicação que você pode iniciar e parar;
- Bancos de Dados Avançados possuem usuários, privilégios de segurança avançados, configurações de rede, ajustes de processamento, memória e armazenamento, além de um complexo sistema de backup e recuperação.



Oracle Database 11g

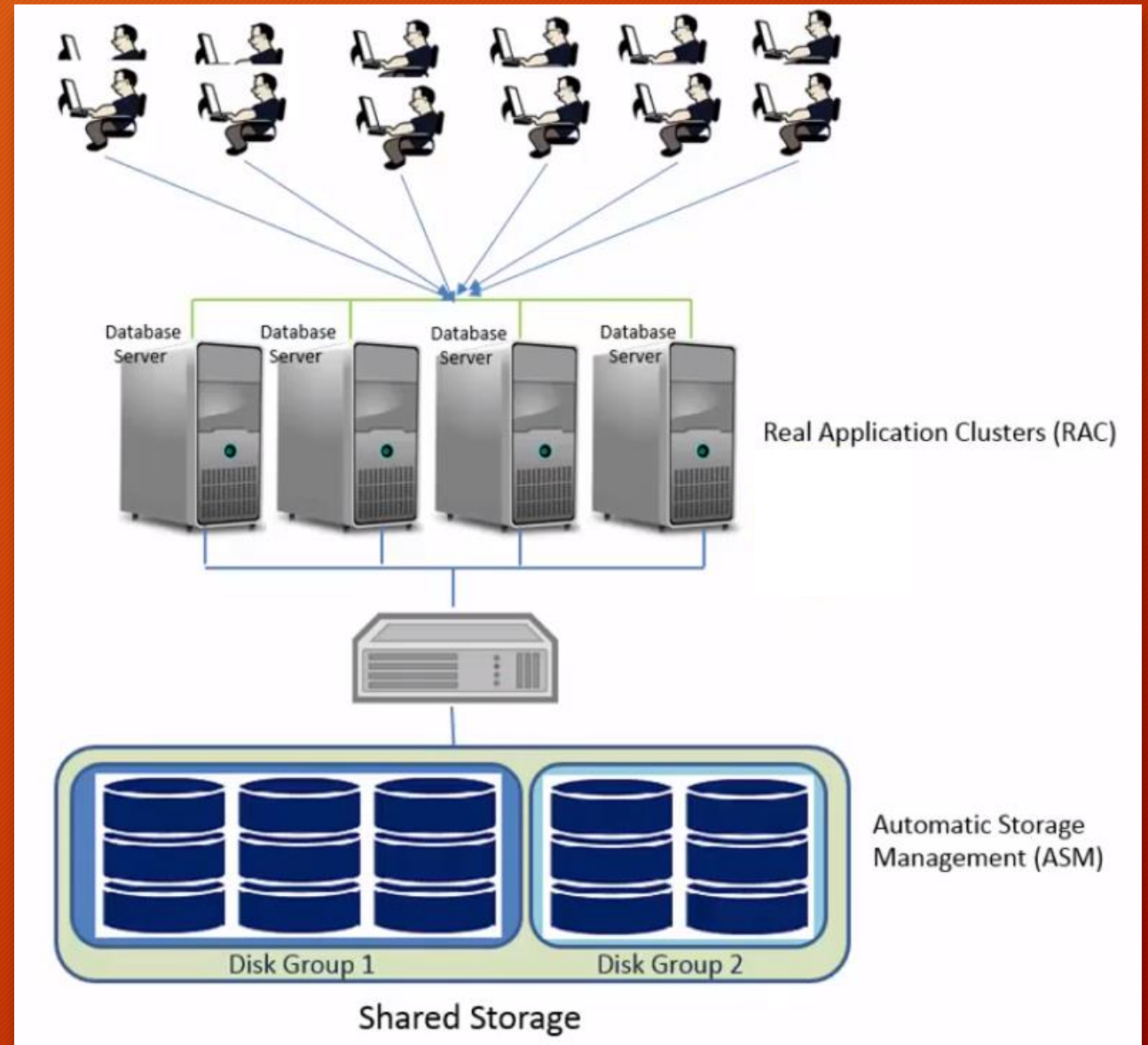
- O “g” do número da versão significa “grid”;
- **Grid Computing** é uma forma de **organizar** os computadores de uma maneira que os seus **recursos** possam ser flexíveis e dinamicamente alocados e acessados. Recursos tais como:
 - **Processador;**
 - **Armazenamento;**
 - **Banco de dados;**
 - **Aplicações**



Oracle Database 11g

Aplicação de conceitos Grid

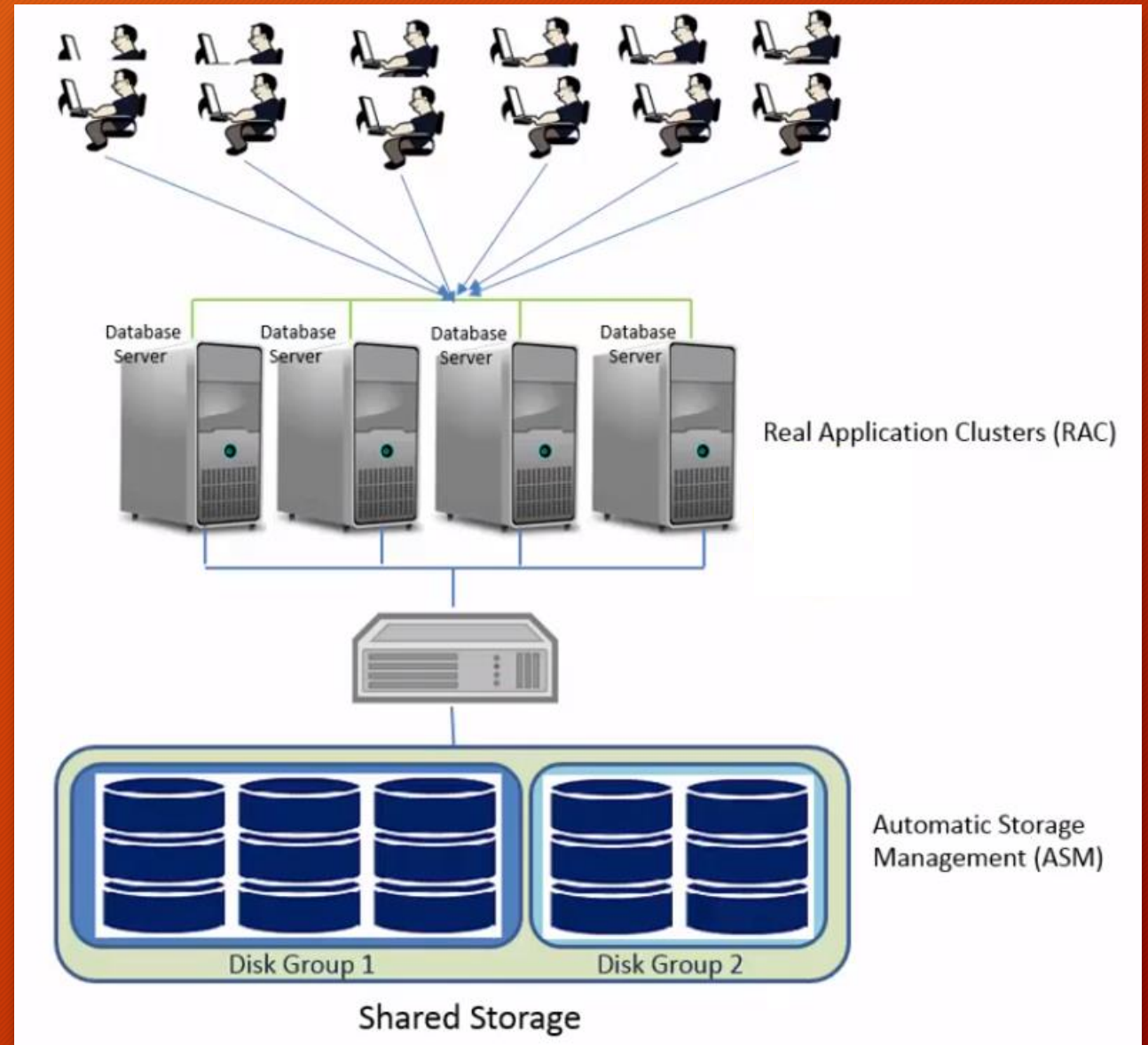
- **RAC - Real Application Cluster** é uma arquitetura que mantém vários servidores de banco de dados;
- O acesso dos usuários é feito de forma **distribuída** e com **balanceamento de carga**;
- Aumenta a **disponibilidade**, já que se um servidor “cair” haverão outros que poderão atender as requisições



Oracle Database 11g

Aplicação de conceitos Grid

- **ASM Automatic Storage Management** é uma arquitetura criada para gerenciar o armazenamento de dados;
 - Os arquivos que compõe o banco de dados são armazenados em múltiplos discos, que garante a disponibilidade;
 - Os servidores podem acessar qualquer disco, já que os dados encontram-se espelhados, o que melhora o desempenho;
 - Os discos podem ser adicionados ou removidos com o banco de dados em funcionamento.



Oracle Database 12c

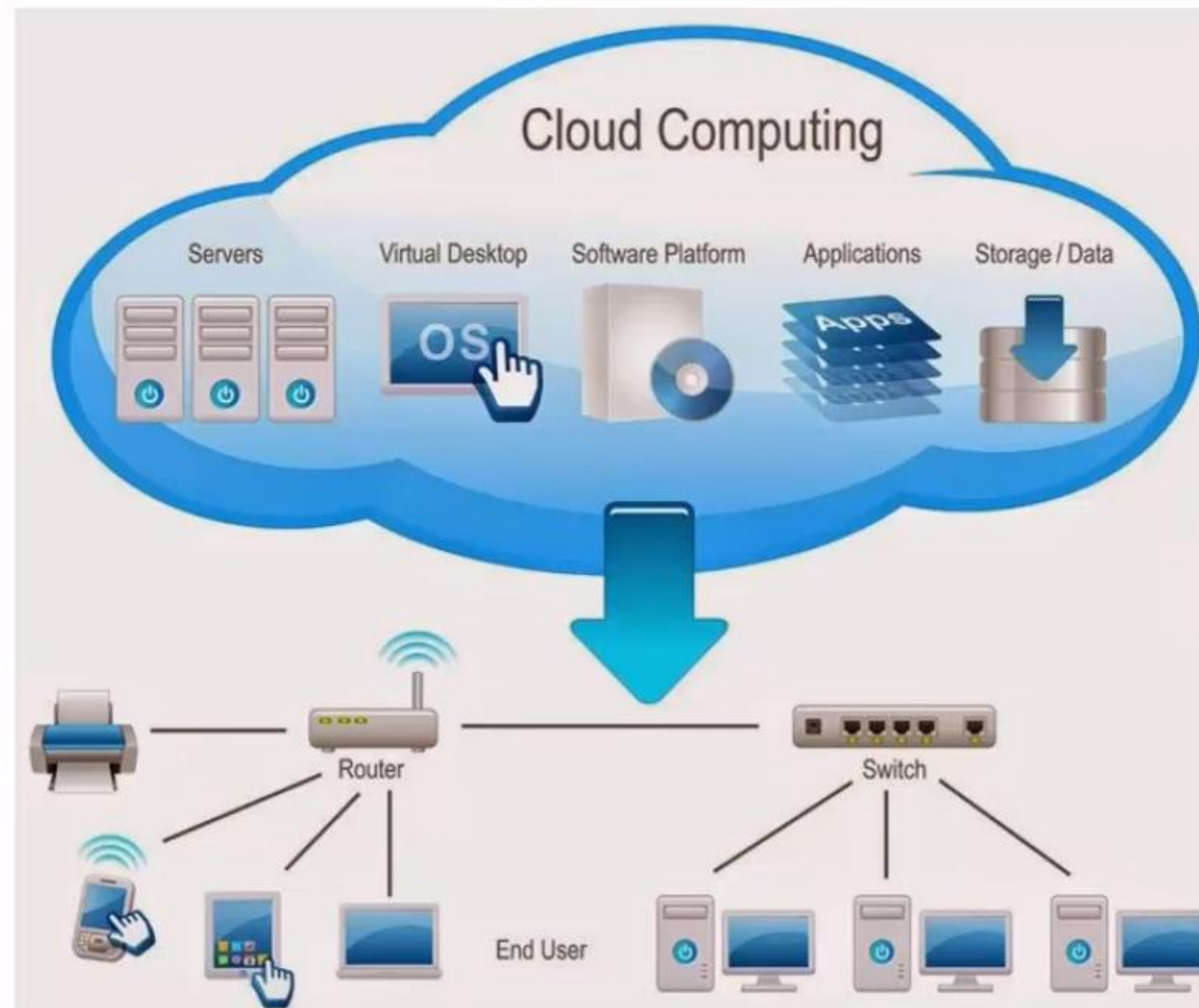
- O “c” do número da versão significa “cloud”;
- A partir da versão 12 o banco de dados Oracle tem suporte para **computação em nuvem**, disponibilizando serviços tais como:
 - Servidores;
 - Máquinas virtuais;
 - Plataformas de software;
 - Aplicações;
 - Armazenamento...



Oracle Database 12c

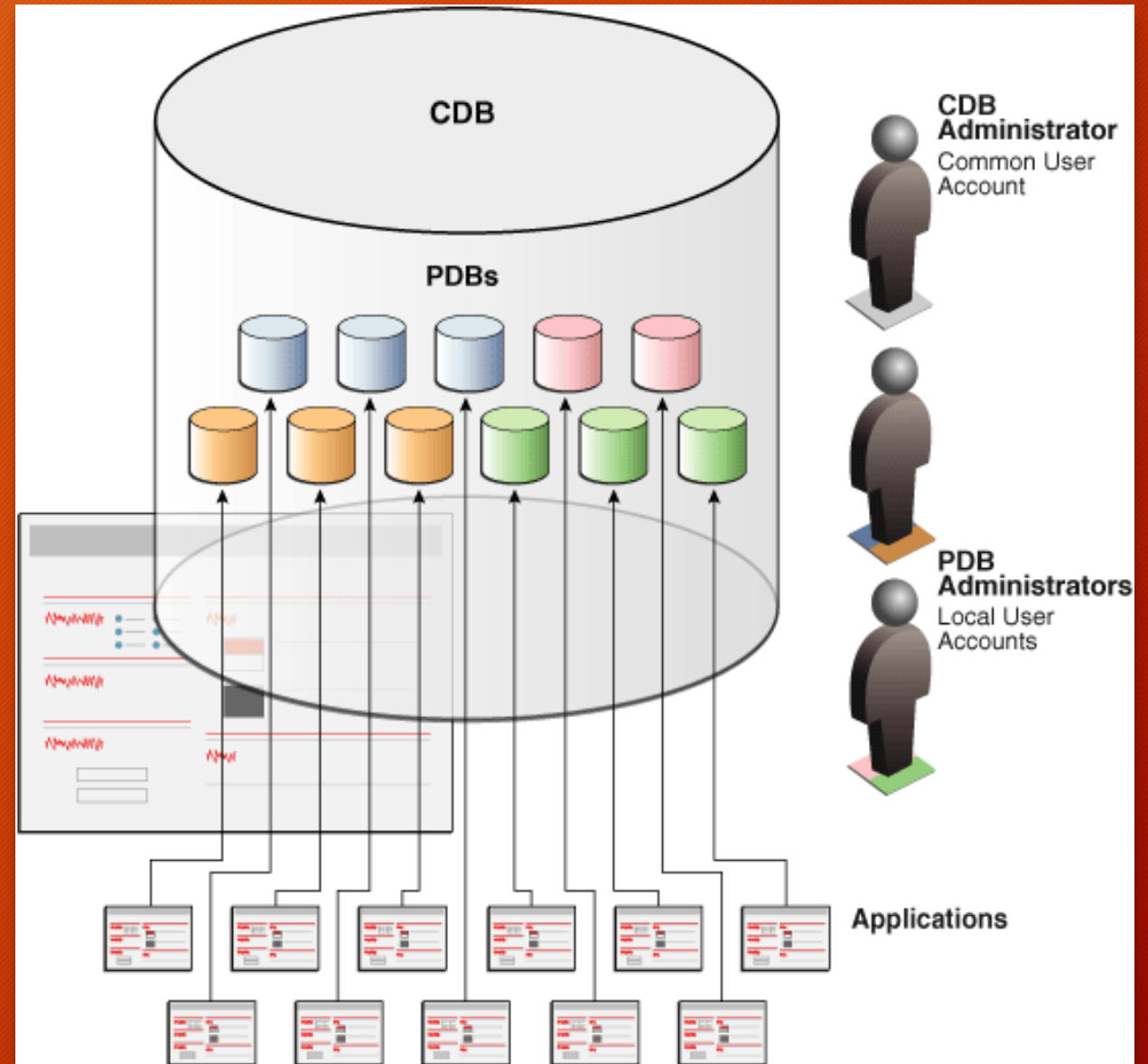
- Todos os serviços ficam disponíveis na rede e as empresas podem pagar para usar estes serviços;
- Os recursos podem ser **ampliados** ou **diminuídos** de acordo com a **demanda** da organização;
- Por exemplo, durante a “Black Friday” uma loja online pode aumentar sua capacidade de processamento e acesso automaticamente;
- O custo ao provedor será de acordo com o uso dos recursos disponíveis.

CLOUD Computing



Oracle Database 12c Multitenant Architecture

- Antes da versão 12c podíamos criar apenas **Container Data Base** (CDB);
- Cada banco de dados tinha sua própria estrutura como memória e arquivo de banco de dados;
- Mesmo com o gerenciamento eficiente de memória e disco havia a duplicação de memória e objetos do banco de dados;
- Com a arquitetura “*Multitenant*” é possível utilizar a mesma estrutura de recursos com vários banco de dados chamados **Pluggable Data Base** - PDB;
- É possível atualizar a versão do banco para apenas alguns banco de dados;
- É possível desconectar um PDB de um CDB e conectá-lo a outro, por exemplo, que tenha a versão correta do banco.



Oracle Database 12c

Undo e Redo

Undo

- Significa reverter a ação de uma ação recente. Por exemplo, podemos reverter a exclusão de um registro em uma tabela;
- O Oracle grava todos os dados que estão prestes a mudar como “**undo**”. Esta informação permite ao Oracle **desfazer** mudanças em caso de “**rollback**”.



Oracle Database 12c

Undo e Redo

Redo

- Significa fazer algo novamente;
- O Oracle grava todas as mudanças efetuadas no banco de dados. Se houver algum problema no banco, esta informação permite ao Oracle refazer a mudança.

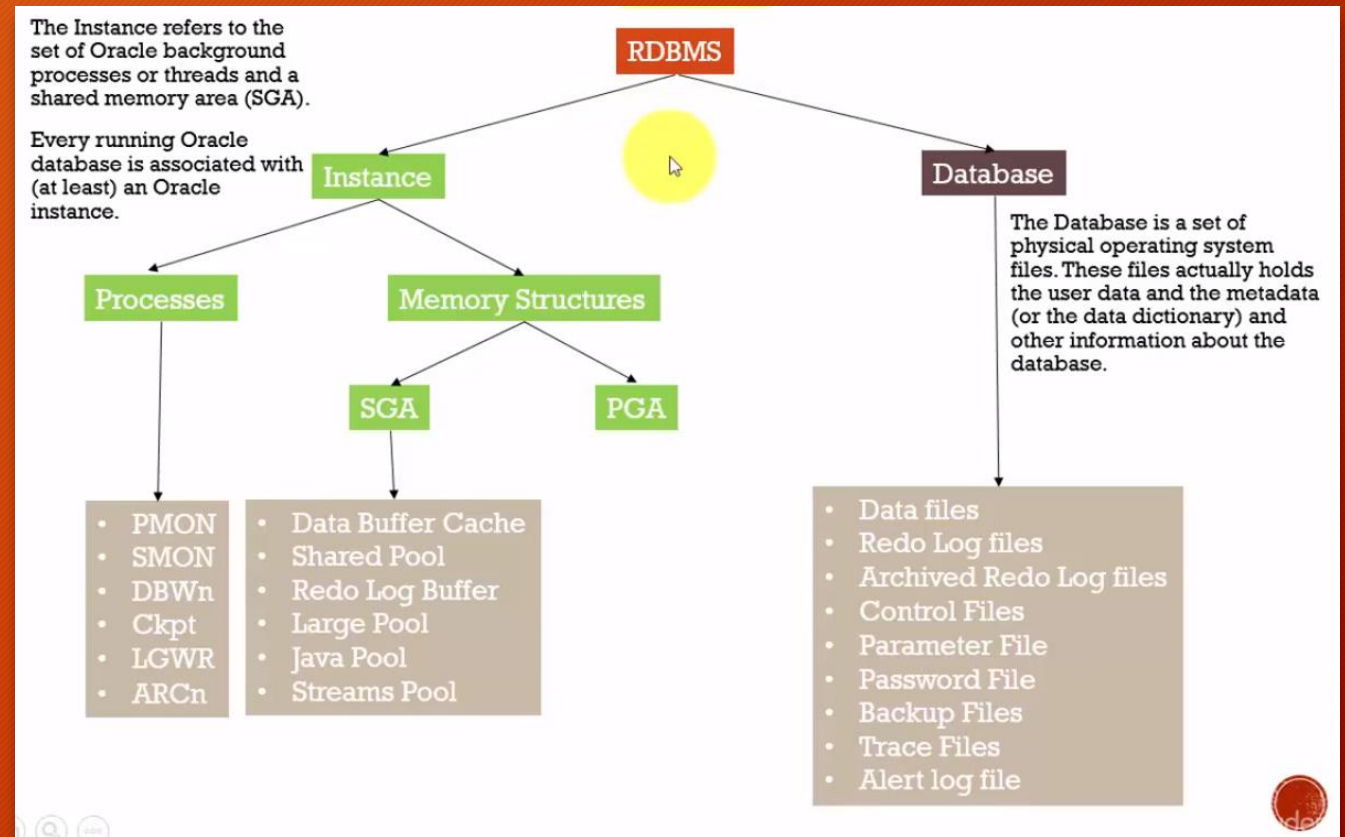


Oracle Database 12c

Instância e Banco de Dados

Instância

- Refere-se ao conjunto de processos do Oracle e compartilhamento de memória (SGA - System Global Área);
- Cada banco de dados rodando no Oracle database está associado com pelo menos uma instância Oracle.

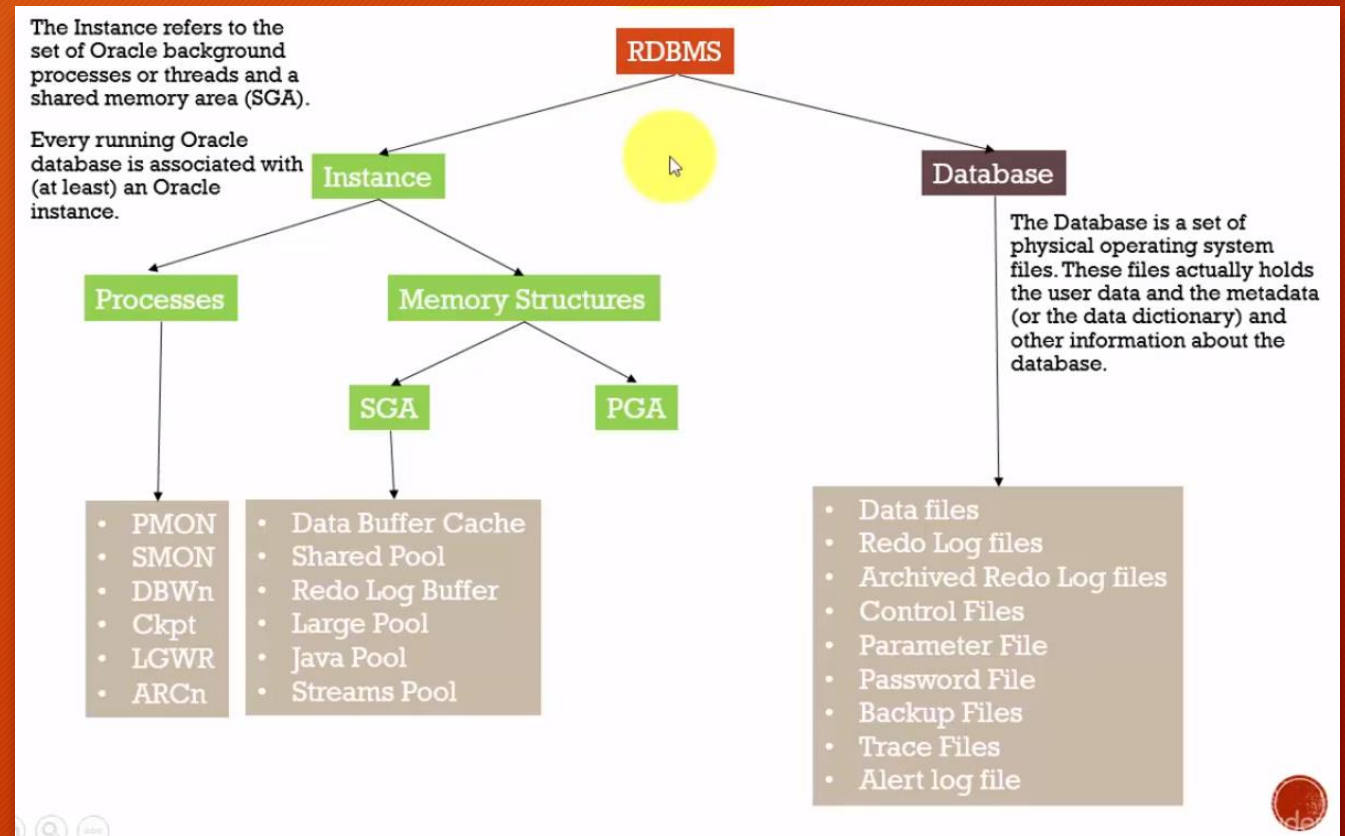


Oracle Database 12c

Instância e Banco de Dados

Banco de dados

- Refere-se ao conjunto de arquivos físicos no sistema operacional;
- Cada arquivo mantém os dados de usuário, dicionário de dados e outras informações sobre o banco de dados.



Criação do banco de dados manualmente

- Criar o arquivo **initprod.ora** com os seguintes parâmetros:
- **db_name**: determina o nome do banco de dados;
- **memory_target**: especifica o tamanho de memória utilizável;
- **control_files**: arquivo que contém entradas que descrevem a estrutura do banco de dados (como seu nome, data de criação, localização dos arquivos de dados, etc);
- **diagnostic_dest**: especifica o diretório onde diagnósticos para uma instância estão localizados.

Criação do banco de dados manualmente

- **O que é um Control File:**
- Cada banco de dados Oracle possui um control file;
- Um control file é um pequeno arquivo binário que armazena a estrutura física do banco de dados, que inclui:
 - Nome do banco de dados,;
 - Nomes e localizações de arquivos de dados (datafiles) associados e log online de arquivos redo;
 - Data de criação do banco de dados;
 - Número de sequência do log atual;
 - Informações de checkpoint.

Criação do banco de dados manualmente

- **O que é um Control File:**

- O control file deve estar **disponível** para **escrita** pelo banco de dados Oracle sempre que o banco de dados é aberto;
- Sem o control file, o banco de dados **não pode ser montado** e a recuperação se torna muito difícil;
- O control file de um banco de dados Oracle é **criado ao mesmo tempo que o banco de dados**;
- Por padrão, pelo menos **uma cópia do control file** é criado durante a criação do banco de dados.
- Você **deve criar uma ou mais cópias do control file** durante a criação do banco de dados.

Criação do banco de dados manualmente

- **Estrutura de memória do banco de dados Oracle**

- Quando uma instância é iniciada, o Oracle database aloca uma área de memória e inicia processos em background. A área de memória armazena informações tais como:
 - Código do programa;
 - Informação sobre cada sessão conectada;
 - Informações necessárias durante a execução do programa, por exemplo o estado de uma consulta das quais linhas foram buscadas;
 - Informações tal como bloqueio de dados que está compartilhado e comunicando-se entre processos;
 - Dados em cache, tais como dados bloqueados e registros redo, que também existem no disco.

Criação do banco de dados manualmente

- **Estrutura de memória do banco de dados Oracle**
- A estrutura básica de memória associadas ao Oracle database incluem:
 - **System Global Area (SGA)**, que é um grupo de estruturas de memória compartilhada conhecidas como SGA componentes, que contém dados e informações de controle para uma instância de banco de dados Oracle;
 - Todos os servidores e processos em segundo plano compartilham a SGA;
 - Exemplos de dados armazenados na SGA incluem cache de dados bloqueados e áreas de SQL compartilhados.

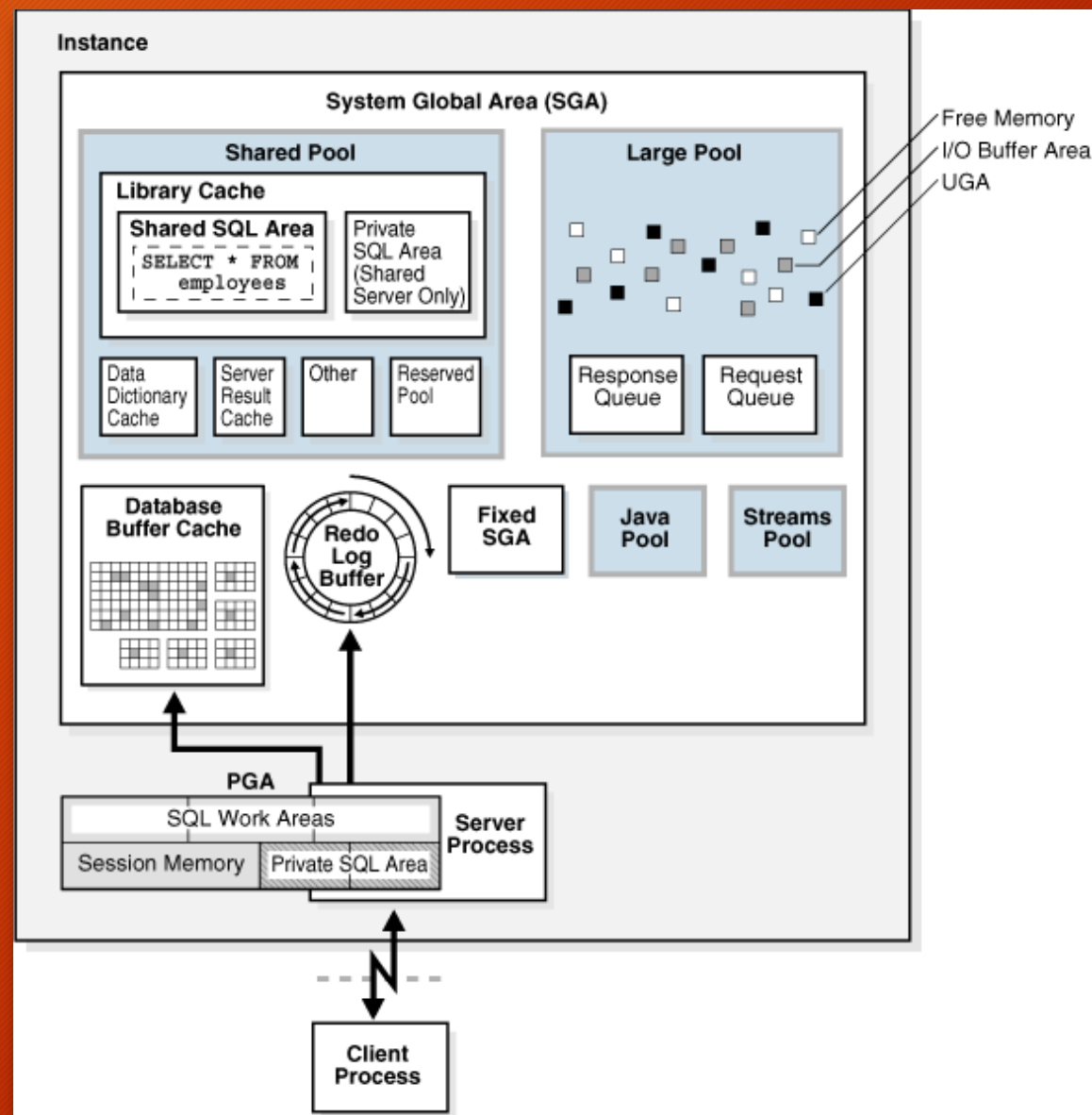
Criação do banco de dados manualmente

- **Estrutura de memória do banco de dados Oracle**
- A estrutura básica de memória associadas ao Oracle database incluem:
 - **Program Global Area (PGA)**, é uma região de memória **não compartilhada** que contém dados e informações de controle **exclusivamente para uso por um processo Oracle**. O PGA é **criado durante a inicialização** de processo Oracle;
 - Um PGA **existe para cada processo de servidor e processos de segundo plano**.

Criação do banco de dados manualmente

Estrutura de memória do banco de dados Oracle

- A estrutura básica de memória associadas ao Oracle database incluem:
 - **User Global Area (UGA)** é a parte da memória associada com a sessão de um usuário;
 - **Software Code Area** é a parte da memória usada para armazenar código que está sendo executado ou poderá ser executado;
 - Código do banco de dados Oracle é armazenado em uma área que está tipicamente em uma localização diferente dos programas de usuário - uma localização mais protegida.



Criação do banco de dados manualmente

- Exemplo do arquivo **initprod.ora**:

```
db_name=prod
memory_target=500m
control_files='/disk1/prod/control/control01.ctl','/disk1/prod/control/control02.ctl'
diagnostic_dest='/disk1/prod/diag'
compatible=11.2.0
```

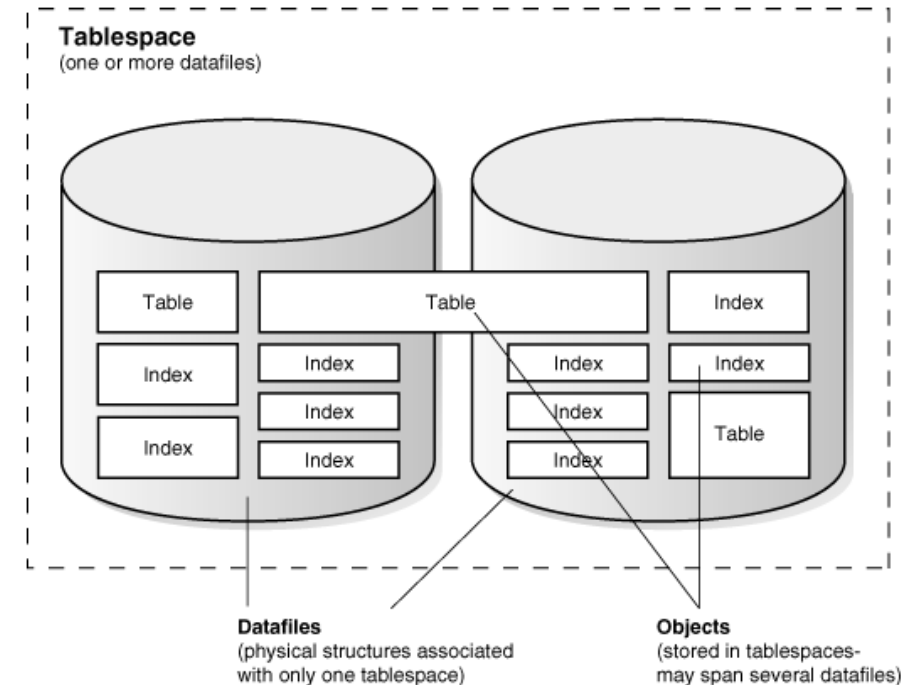
Criação do banco de dados manualmente

- Um banco de dados Oracle pode ser criado de forma manual ou utilizando o assistente de criação de banco de dados “DBCA”;
- Para ambas as formas de criação do banco de dados será necessário informar basicamente os seguintes parâmetros:
 - Nome do banco de dados;
 - Datafile “system.dbf”;
 - Datafile “sysaux.dbf”;
 - User data tablespace;
 - Undo tablespace;
 - Logfiles.

Criação do banco de dados manualmente

Tablespaces e Datafiles

- O Oracle armazena os dados logicamente em **tablespaces** e fisicamente em **datafiles** associados com o tablespace correspondente, conforme a figura ao lado.

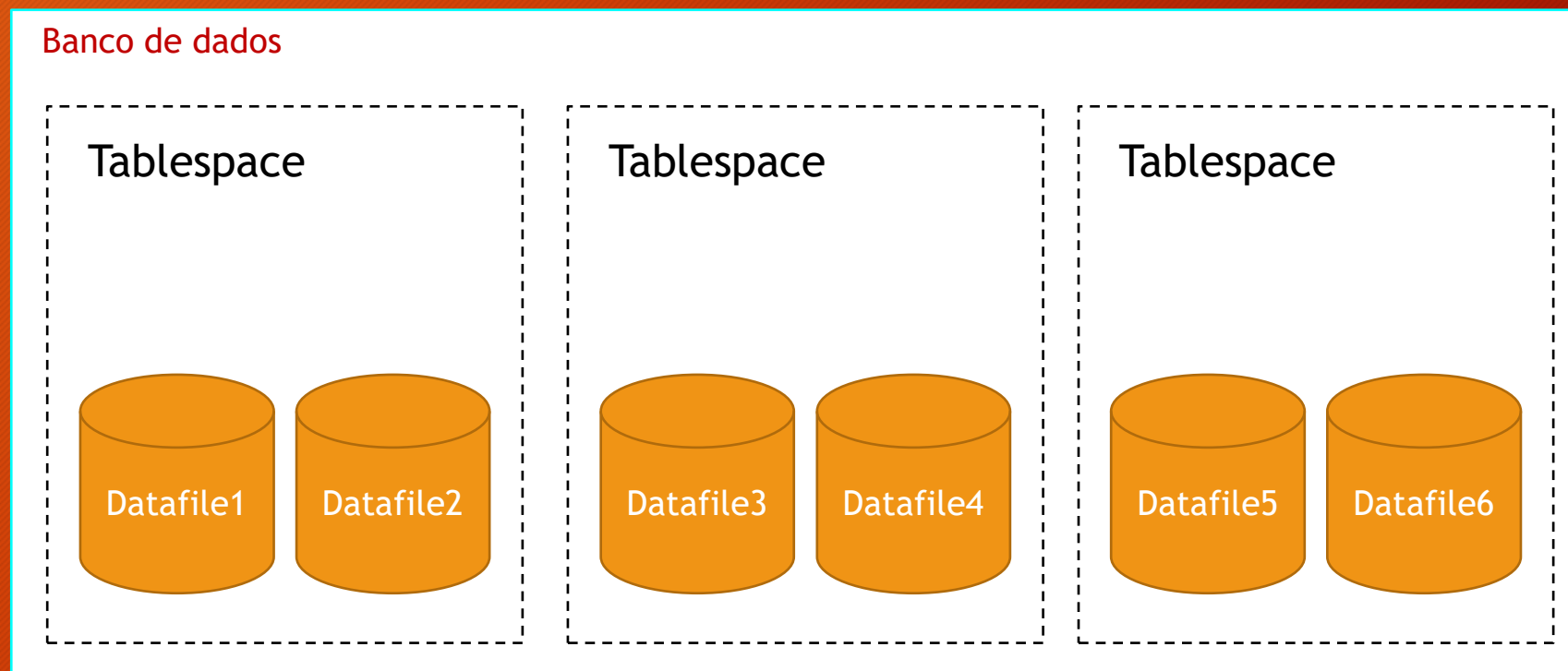
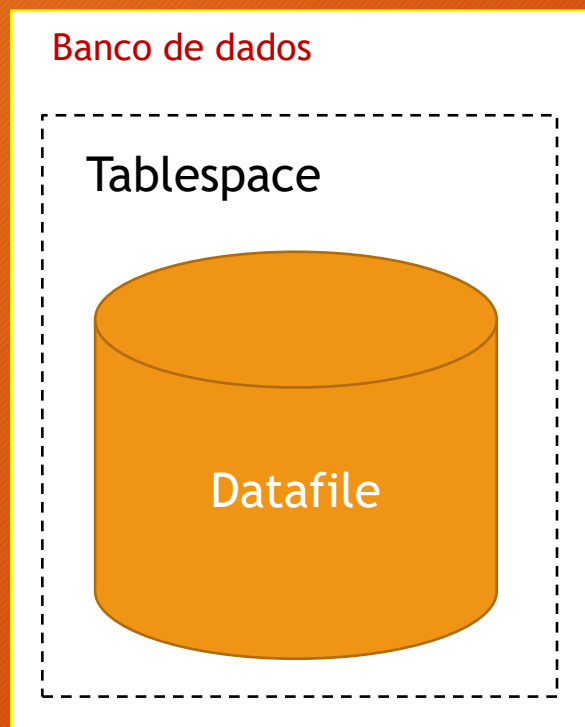


Criação do banco de dados manualmente

- Databases, tablespaces e datafiles são muito parecidos, mas eles possuem diferenças importantes;
- Cada tablespace em um Oracle database consiste de um ou mais arquivos chamados “datafiles”, os quais são estruturas físicas em conformidade com o sistema operacional que estão executando;
- Os dados do banco de dados estão armazenados coletivamente nos datafiles que constituem cada tablespace do banco de dados.

Criação do banco de dados manualmente

- Por exemplo, o banco de dados Oracle **mais simples** deveria ter **um tablespace** e um datafile. Outro banco de dados pode ter três tablespace, e cada um consistindo de dois datafiles, totalizando seis datafiles.



Criação do banco de dados manualmente

- Criar o arquivo **dbcreation.sql** com os seguintes parâmetros:
- **create database**: define o nome do banco de dados;
- **datafile**: especifica o tamanho de memória utilizável;
- **control_files**: arquivo que contém entradas que descrevem a estrutura do banco de dados (como seu nome, data de criação, localização dos arquivos de dados, etc);
- **diagnostic_dest**: especifica o diretório onde diagnósticos para uma instância estão localizados.

Criação do banco de dados manualmente

Exemplo de arquivo de criação do banco:

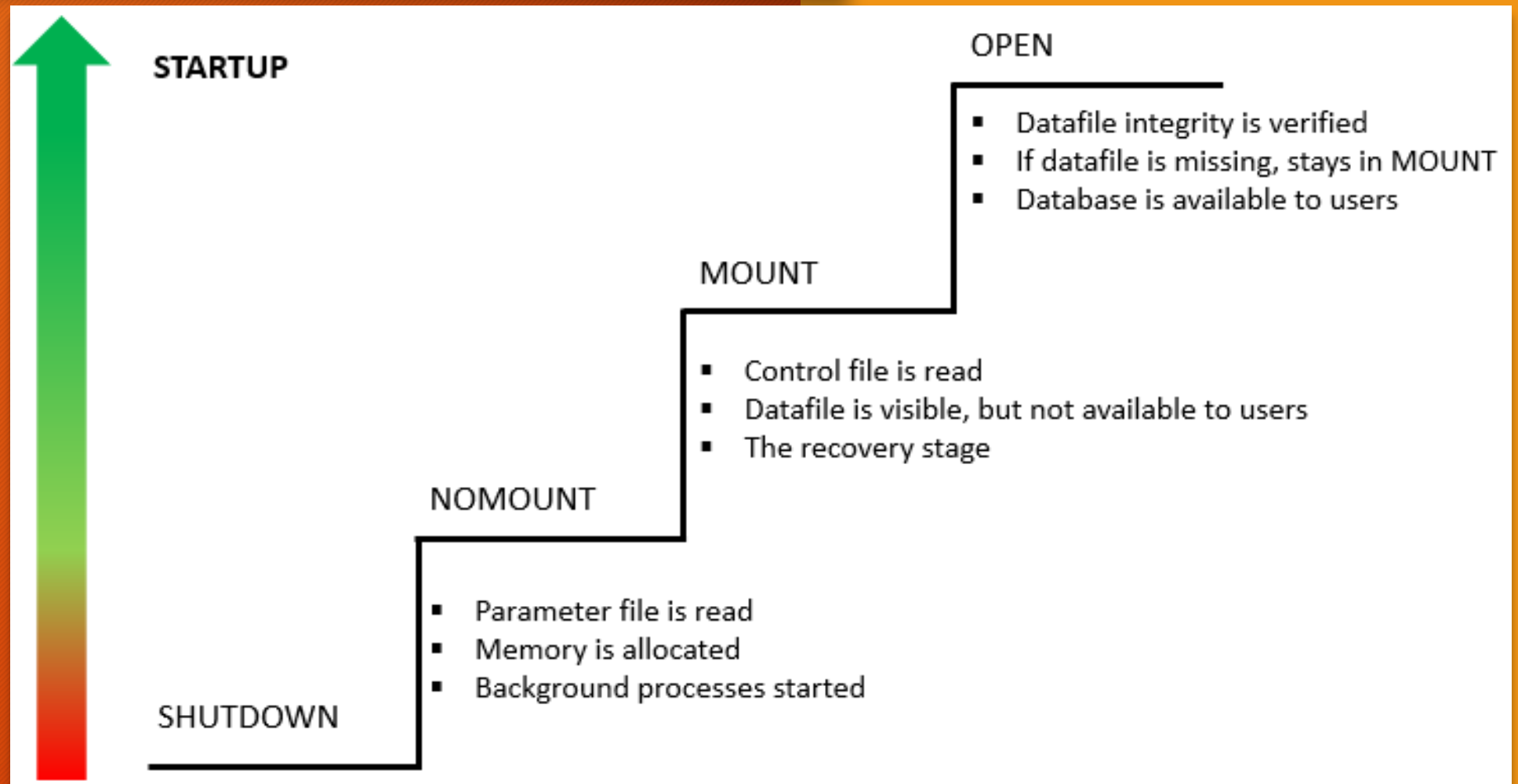
```
create database prod
datafile '/disk1/prod/data/system.dbf' size 300m autoextend on
sysaux datafile '/disk1/prod/data/sysaux.dbf' size 300m autoextend on
default tablespace user_data datafile '/disk1/prod/data/user01.dbf' size 500m
undo tablespace undotbs datafile '/disk1/prod/data/undotbs.dbf' size 100m
logfile group 1 '/disk1/prod/redo/redo1.rdo' size 100m,
        group 2 '/disk1/prod/redo/redo2.rdo' size 100m;
```

Passos de inicialização do Oracle

- Se você precisa acessar o banco de dados, a instância correspondente ao banco de dados deve ser iniciada, montada e aberta;
- A inicialização de um banco de dados envolve:
 1. Iniciar uma instância;
 2. Montar o banco de dados;
 3. Abrir o banco de dados



Passos de inicialização do Oracle



Passos de finalização do Oracle

- Há diferentes modos para parar um banco de dados Oracle, são eles:
 1. Shutdown immediate;
 2. Shutdown transactional;
 3. Shutdown normal;
 4. Shutdown abort;



Passos de finalização do Oracle

Shutdown Modes

Shutdown Modes	A	I	T	N
Allows new connections	No	No	No	No
Waits until current sessions end	No	No	No	Yes
Waits until current transactions end	No	No	Yes	Yes
Forces a checkpoint and closes files	No	Yes	Yes	Yes

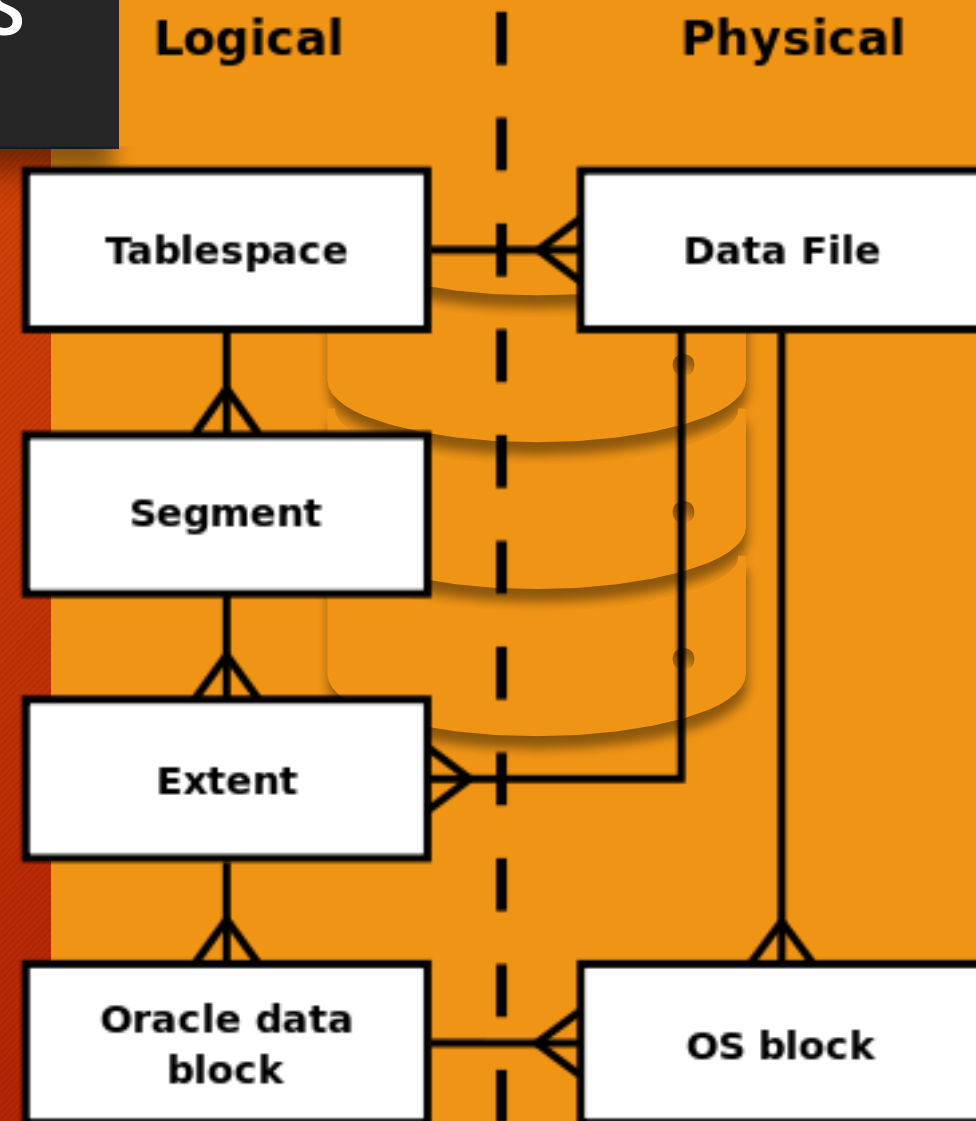
Shutdown modes:

- A = ABORT
- I = IMMEDIATE
- T = TRANSACTIONAL
- N = NORMAL

Como os dados são armazenados

- Um banco de dados Oracle tem a sua estrutura armazenada de duas formas:

- **Física e Lógica**



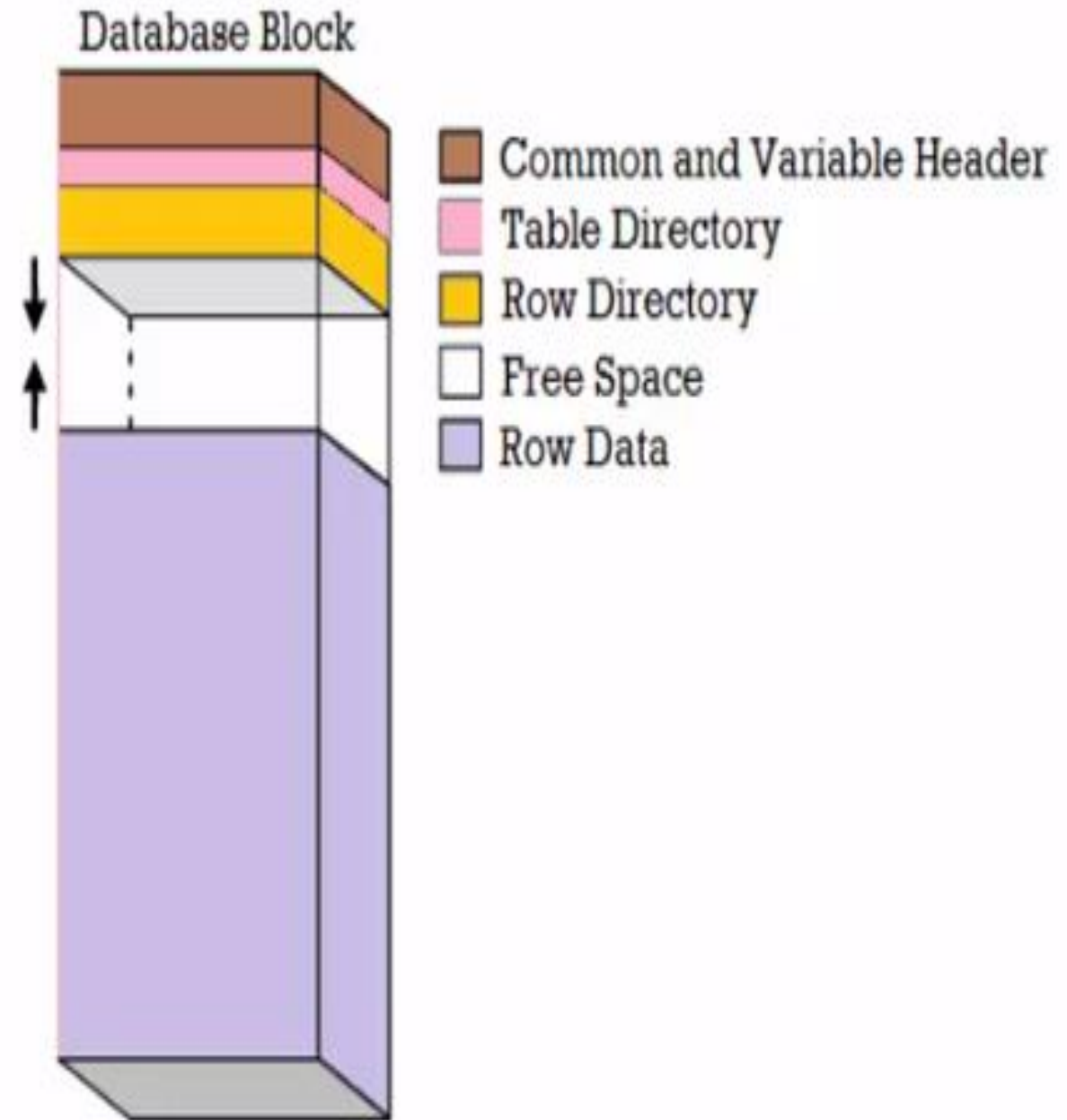
Estrutura Lógica

- O banco de dados Oracle aloca espaço lógico para todos os dados no banco de dados;
- As unidades lógicas do espaço de alocação dos banco de dados são:
 - **Data Blocks;**
 - **Extents;**
 - **Segments;**
 - **Tablespace;**

Estrutura Lógica

Data block

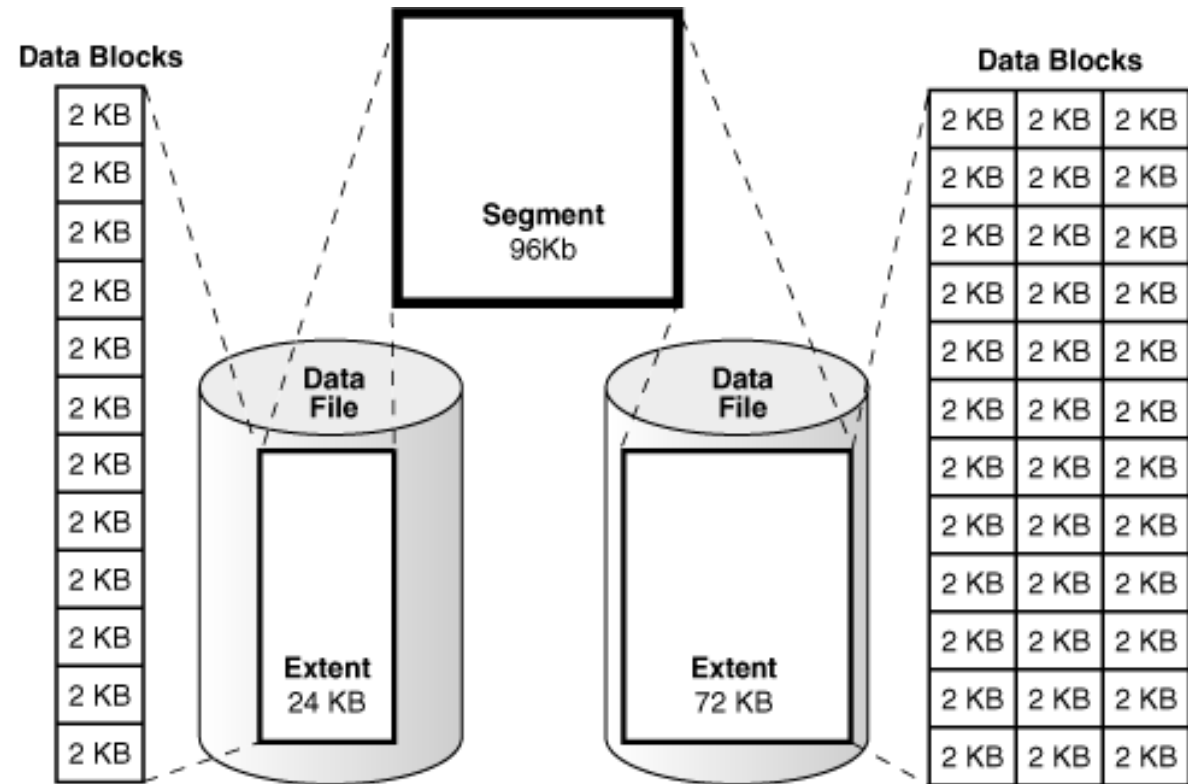
- No menor nível de granularidade o Oracle armazena os dados em blocos de dados;
- Um bloco de dados corresponde a um número específico de bytes do espaço físico de disco, por exemplo 2 KB;
- São a menor unidade de armazenamento que o Oracle pode usar e alocar.



Estrutura Lógica

Extent

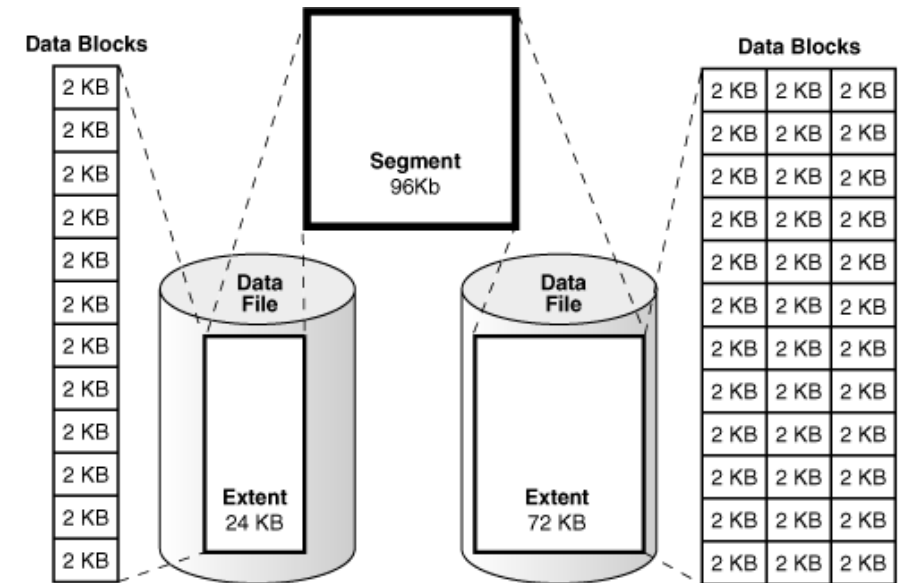
- O extent é um conjunto de blocos de dados logicamente contíguos alocados para armazenar um tipo específico de informação;
- Na figura ao lado, o extent de 24 KB tem 12 data blocks, enquanto que o extent de 72 KB tem 36 data blocks.



Estrutura Lógica

Segment

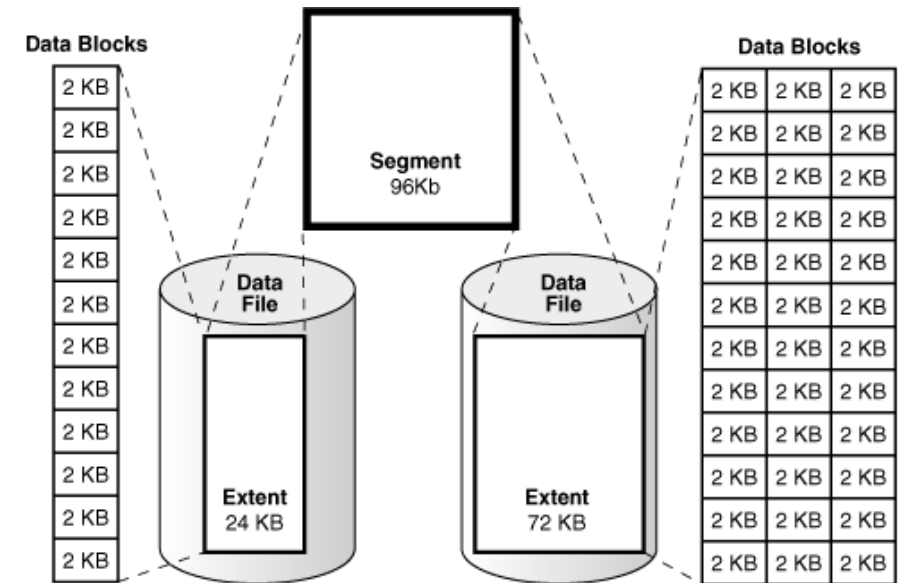
- O segment é um conjunto de extents alocados para um objeto de banco de dados específico, como uma tabela;
- Por exemplo, os dados na tabela de funcionários é armazenado em seu próprio segmento de dados, enquanto que cada índice para empregados é armazenado em seu próprio segmento de índice;
- Cada objeto do banco de dados que consome armazenamento consiste de um segmento.



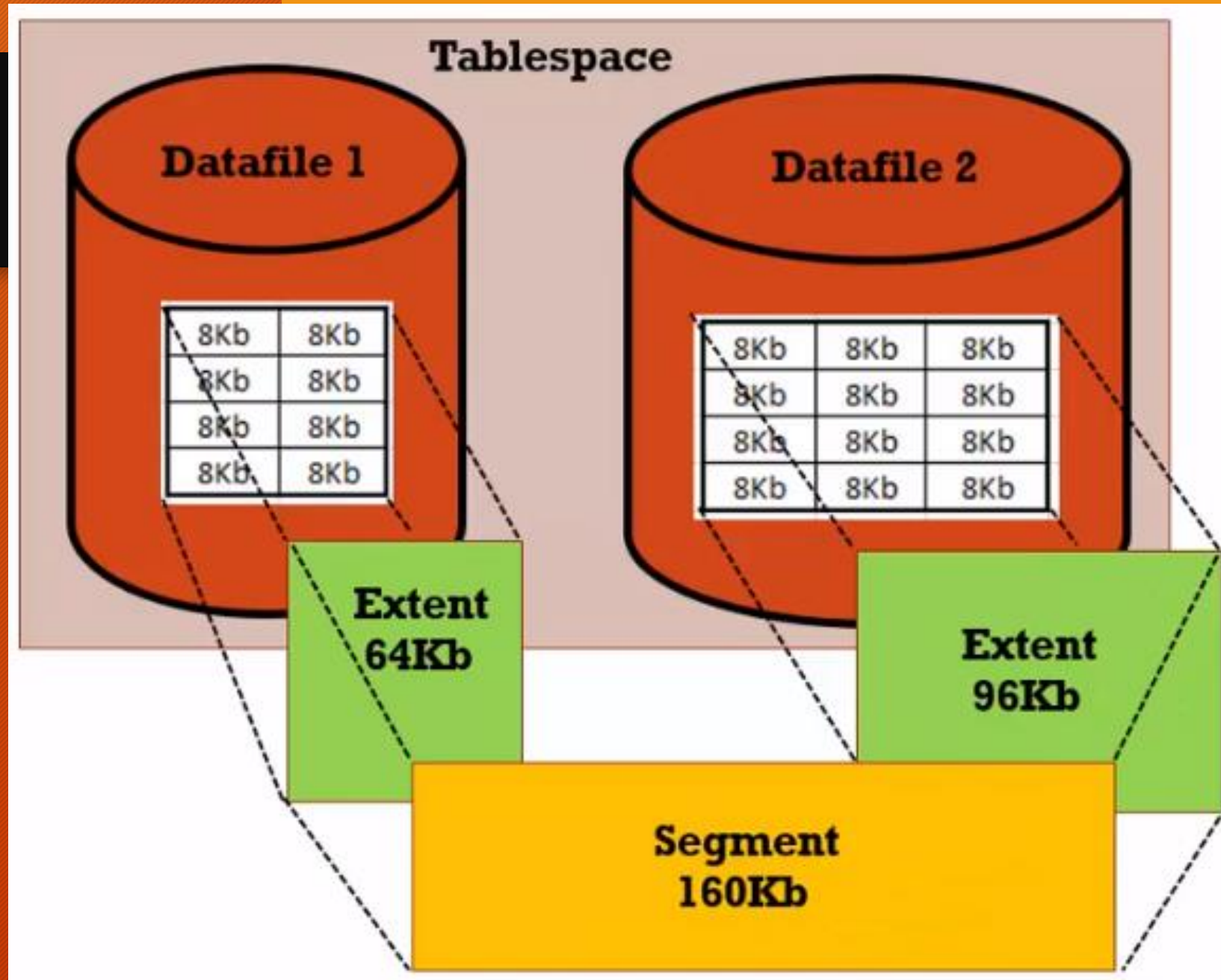
Estrutura Lógica

Segment

- Cada segment pertence a um e somente um tablespace;
- Assim, todos os extents para um segmento são armazenados no mesmo tablespace;
- Dentro de um tablespace, um segment pode incluir extents de múltiplos data files;
- Por exemplo, um extent para um segment pode ser armazenado em user01.dbf, enquanto outro é armazenado em users02.dbf;.

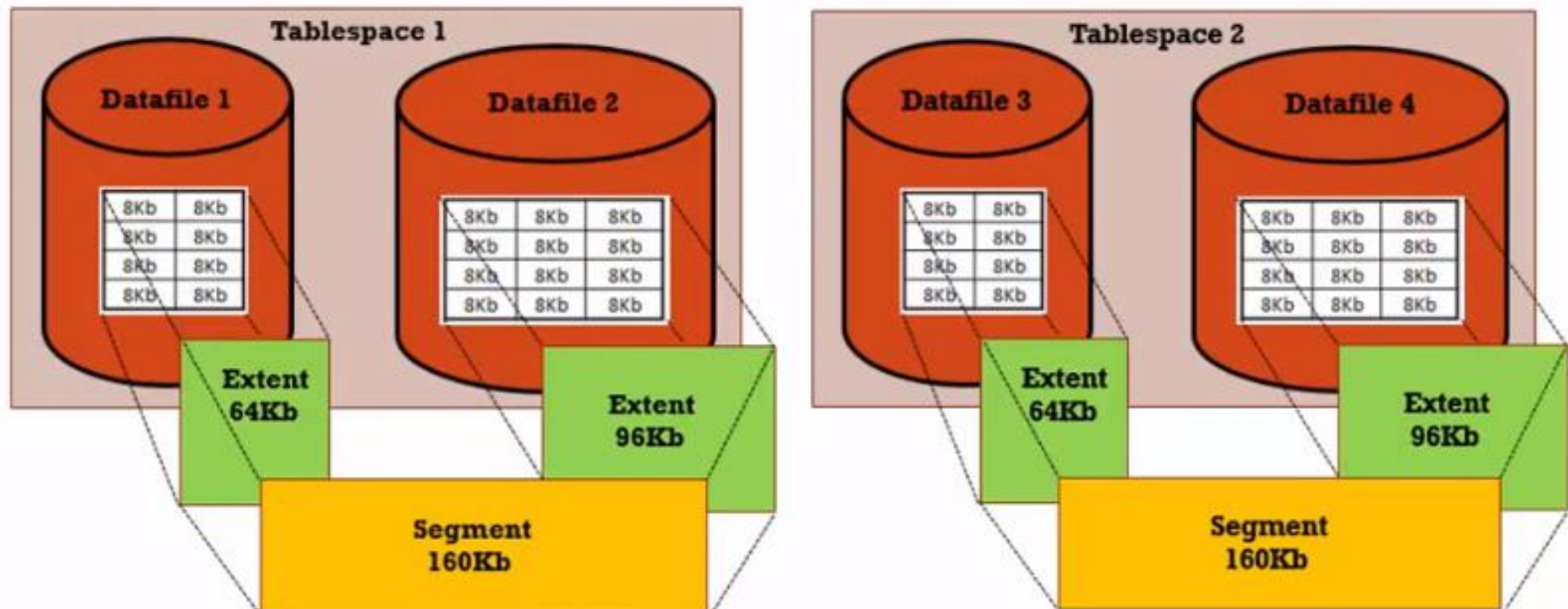


Estrutura Lógica



Estrutura Lógica

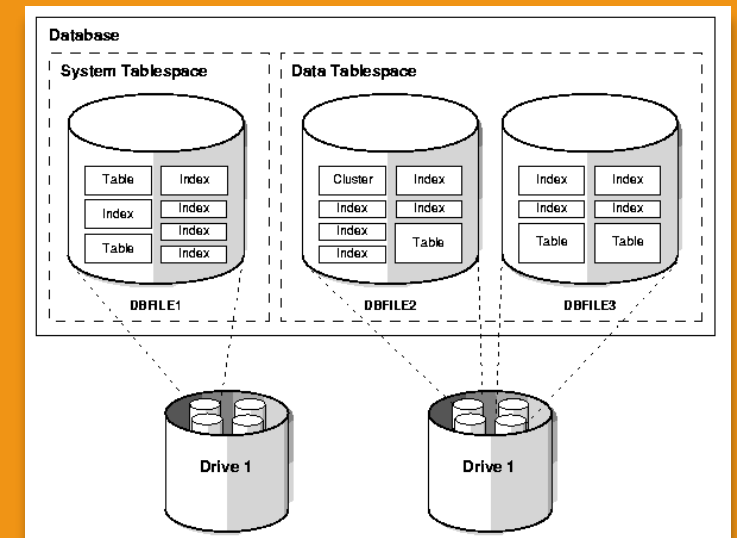
Database



Tablespaces

São containers lógicos para armazenar segmentos. Segmentos são objetos do banco de dados, como tabelas e índices, que consomem espaço de armazenamento. No nível físico, um tablespace é formado por um ou mais data files. Os tablespaces default são:

- SYSTEM
- SYSAUX
- TEMP
- UNDOTBS1
- USERS
- EXAMPLE



Tablespaces

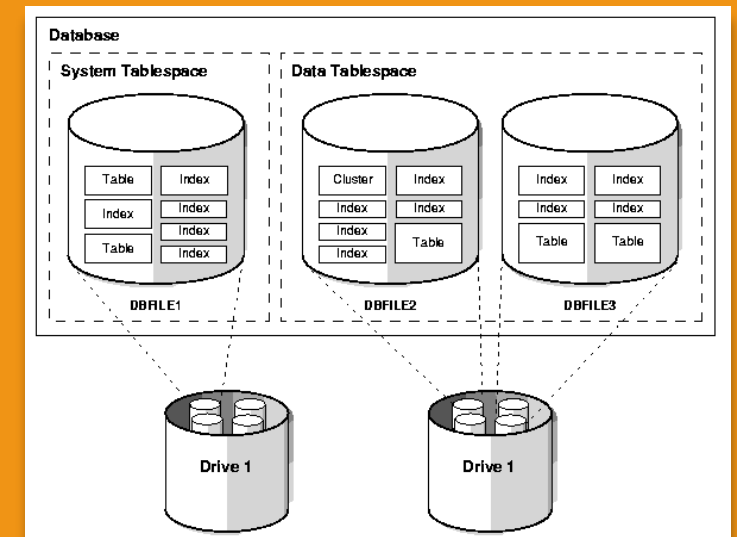
UNDO tablespace

Todos os bancos de dados Oracle precisam de um local para armazenar informações a desfazer, ou seja, esse tablespace contém seus segmentos de reconstrução;

Possui a capacidade de recuperar transações incompletas ou abortadas;

Um segmento de UNDO é usado para salvar o valor antigo quando um processo altera o valor do banco de dados;

Ele armazena a localização dos dados e também os dados da forma como os dados se encontravam antes da modificação;



Tablespaces

UNDO tablespace

Basicamente, os objetivos de UNDO são:

Rollback de Transação: quando uma transação **modifica** uma linha de uma tabela, a imagem original das colunas modificadas é **salva** no segmento de UNDO, e se for feito o rollback da transação, o servidor Oracle **restaurará** os valores originais gravando os valores do segmento de UNDO novamente na linha.



Tablespaces

UNDO tablespace

Basicamente, os objetivos de UNDO são:

Recuperação de Transação: Se ocorrer uma **falha de instância** enquanto houver transações em andamento, o servidor Oracle precisará **desfazer** as alterações não submetidas à **commit** quando o banco de dados for aberto novamente. Esse rollback faz parte da **recuperação da transação**;

Portanto, a recuperação só é possível porque as alterações feitas no segmento de UNDO também são protegidas pelos arquivos de redo log online.



Tablespaces

UNDO tablespace

Basicamente, os objetivos de UNDO são:

Consistência de Leitura: Enquanto houver transações em andamento, outros usuários do banco de dados não deverão ver as alterações não submetidas à commit feitas nessas transações;

Além disso, uma instrução não deverá ver as alterações submetidas à commit após o início da execução dessa instrução;

Os valores antigos (dados de undo) dos segmentos de UNDO também são usados para oferecer aos leitores uma imagem consistente de uma instrução específica.



Tablespaces

SYSTEM tablespace

- É usado pelo Oracle Server para gerenciar o banco de dados;
- Ele contém o dicionário de dados que guarda informações administrativas sobre o banco de dados;
- Nem todos os usuários podem ter acesso a esse tablespace;
- Não podemos renomear, apagar e nem torna-lo offline;
- Normalmente você pode armazenar dados do usuário aqui, mas não é recomendável;
- Deve ser usado puramente pelo Oracle Server para armazenar todas as informações administrativas e tabelas de dicionário de dados.



Tablespaces

SYSAUX tablespace

- Que significa “auxiliar do sistema”;
- É um tablespace auxiliar para o SYSTEM tablespace;
- É obrigatório;
- Alguns dos objetos que antes eram armazenados no SYSTEM tablespace agora são armazenados em SYSAUX;
- Possui as mesmas condições de um SYSTEM tablespace, ou seja, não pode ser renomeado, apagado e nem colocado em offline;



Tablespaces

TEMP tablespace

- Utilizado para armazenar todos os dados temporários sempre que você executa uma instrução SQL que requer armazenamento e leitura, como operações de ordenação, funções matemáticas, criação de índices.



Tablespaces

USERS tablespace

- Utilizado para armazenar todos os dados criados pelos usuários;
- Normalmente, durante a criação da tabela o DBA atribui um tablespace para a tabela que precisa ser armazenada;
- Se o DBA não indicar qualquer tablespace, por default todos os dados relacionados aquela tabela são armazenados no USERS tablespace.



Tablespaces

EXAMPLE tablespace

- Este tablespace contém todas as tabelas de exemplo que são instalados quando você cria o banco de dados;
- Se você não instalar as tabelas de exemplo este tablespace não será criado.



Tablespaces

ONLINE/OFFLINE

- Quando o tablespace está executando e o Oracle pode ler ou escrever no tablespace dizemos que ele está **ONLINE**.
- O tablespace está **OFFLINE** quando o Oracle não pode ler ou escrever no tablespace;
- Para executar tarefas de backup, recovery, remoção ou movimentação de um tablespace ou data file sem fechar o banco de dados, colocamos o tablespace em modo **OFFLINE**.



Tablespaces

Criação de tablespace no Oracle Database

```
create tablespace tbs1  
datafile '/disk1/dev/data/data01.dbf'  
size 50m  
autoextend on  
next 512k  
maxsize 250m;
```

Tablespaces

Aumentar o tamanho de um tablespace existente

```
alter tablespace tbs1  
add datafile '/disk1/dev/data/data02.dbf'  
size 50m  
autoextend on  
next 512k  
maxsize 250m;
```

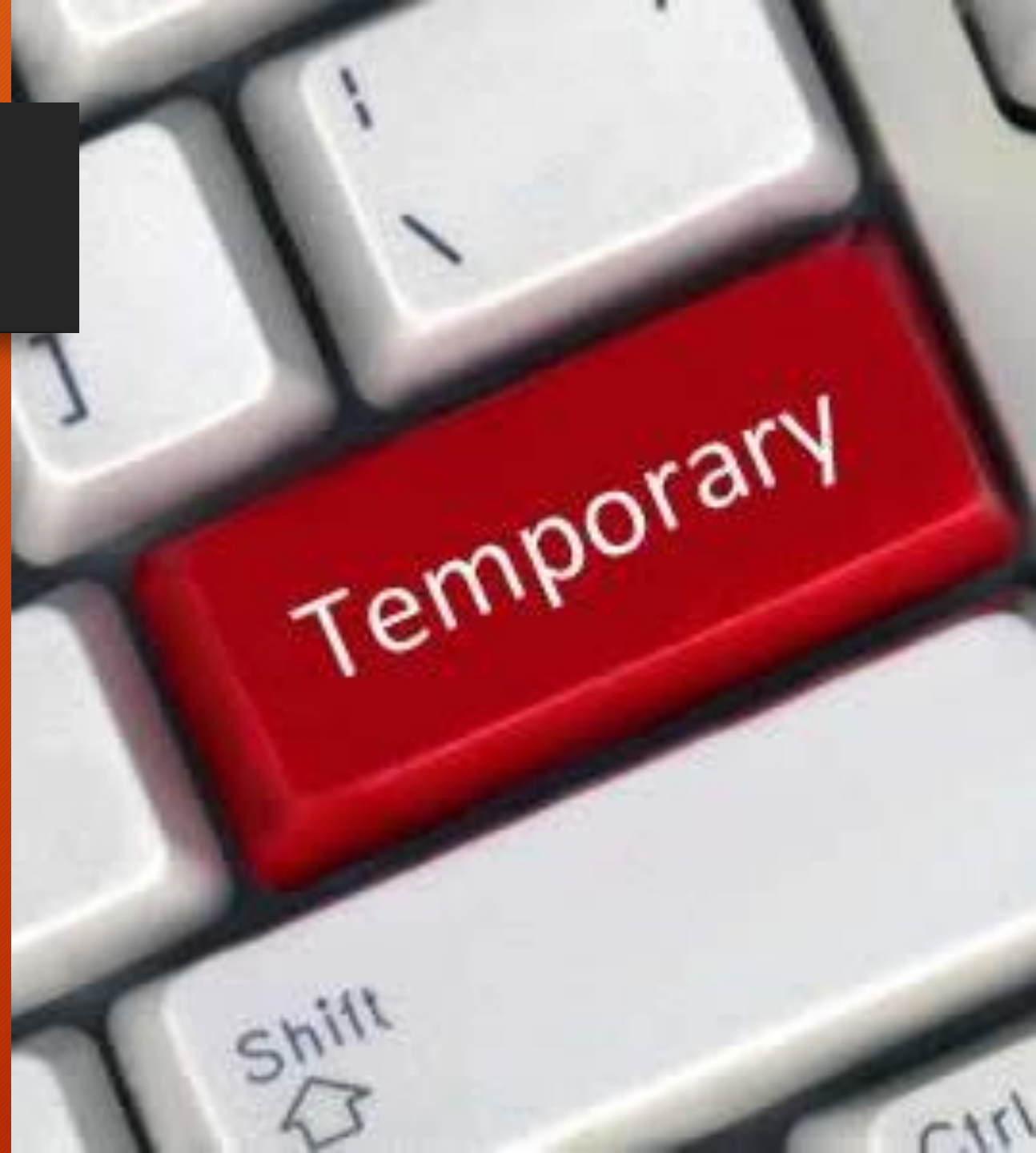

Tablespaces

Aumentar o tamanho de um banco de dados

```
alter database  
datafile '/disk1/dev/data/data02.dbf'  
resize 100m;
```

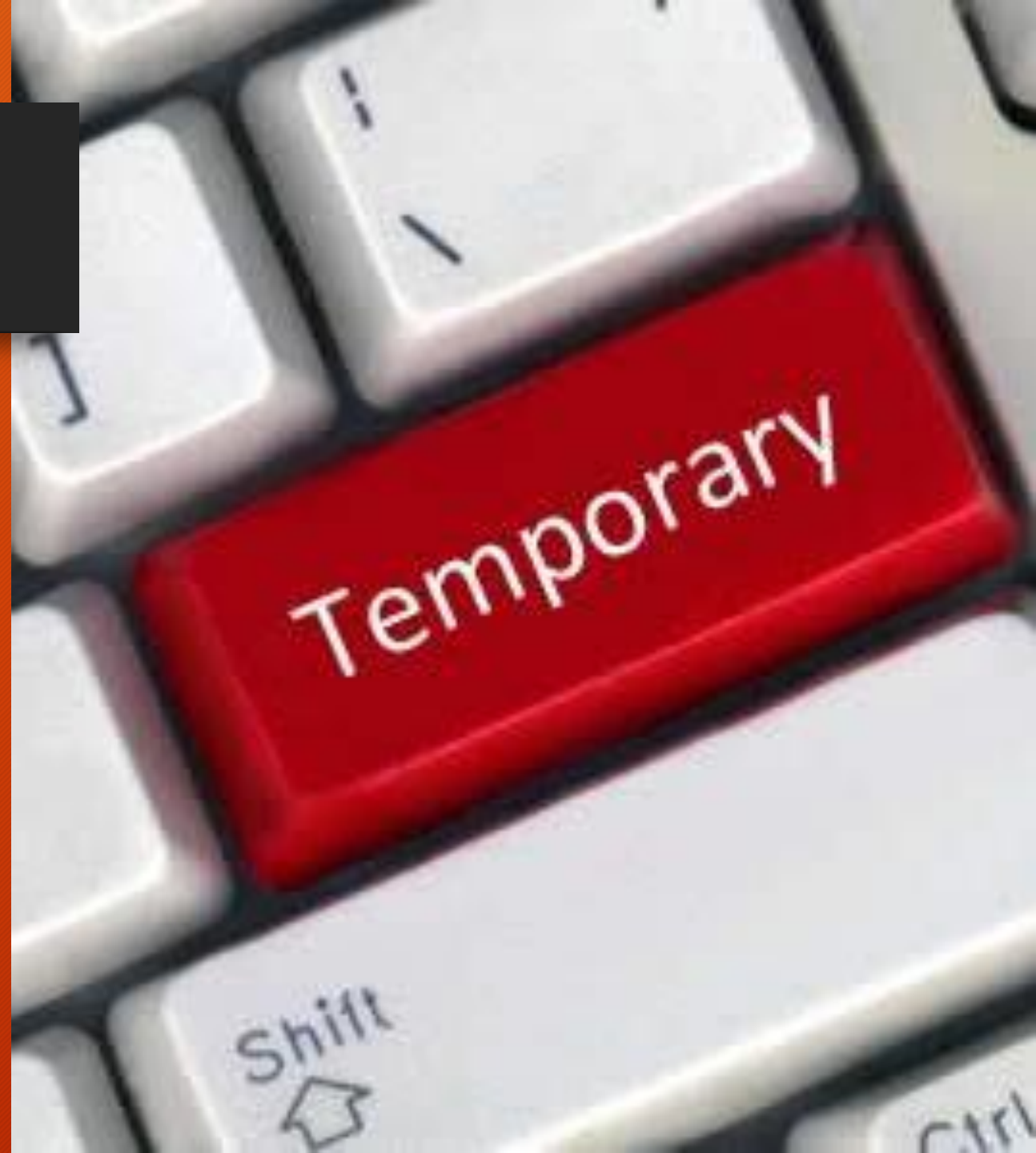
Tablespace temporário

Os tablespaces temporários são utilizados para manusear operações do banco de dados, e são usados também para armazenar tabelas locais temporariamente.



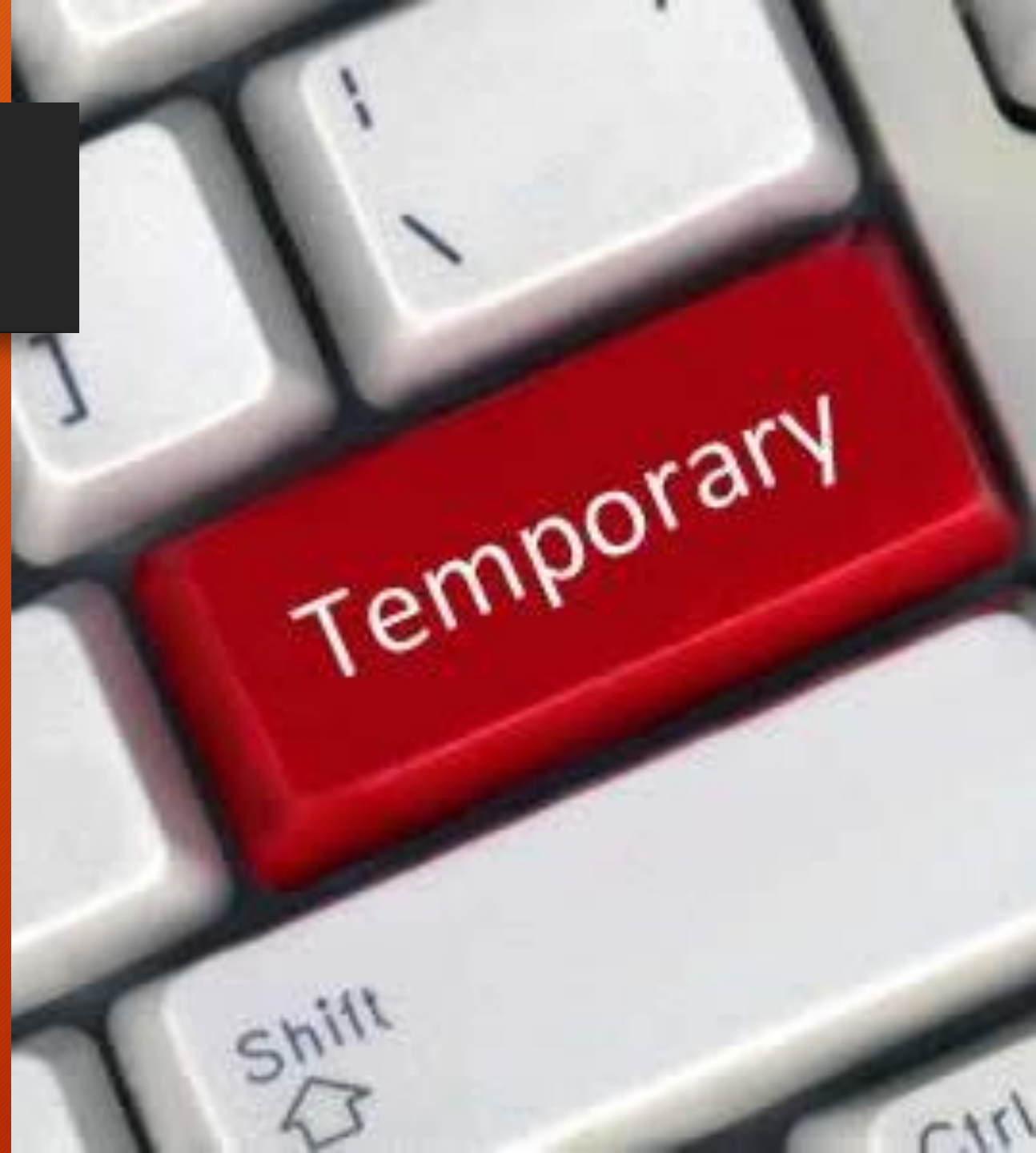
Tablespace temporário

Por exemplo, quando você junta duas tabelas muito grandes o Oracle pode não possuir memória RAM suficiente, então ele aloca o resultado em um Temporary Tablespace



Tablespace temporário

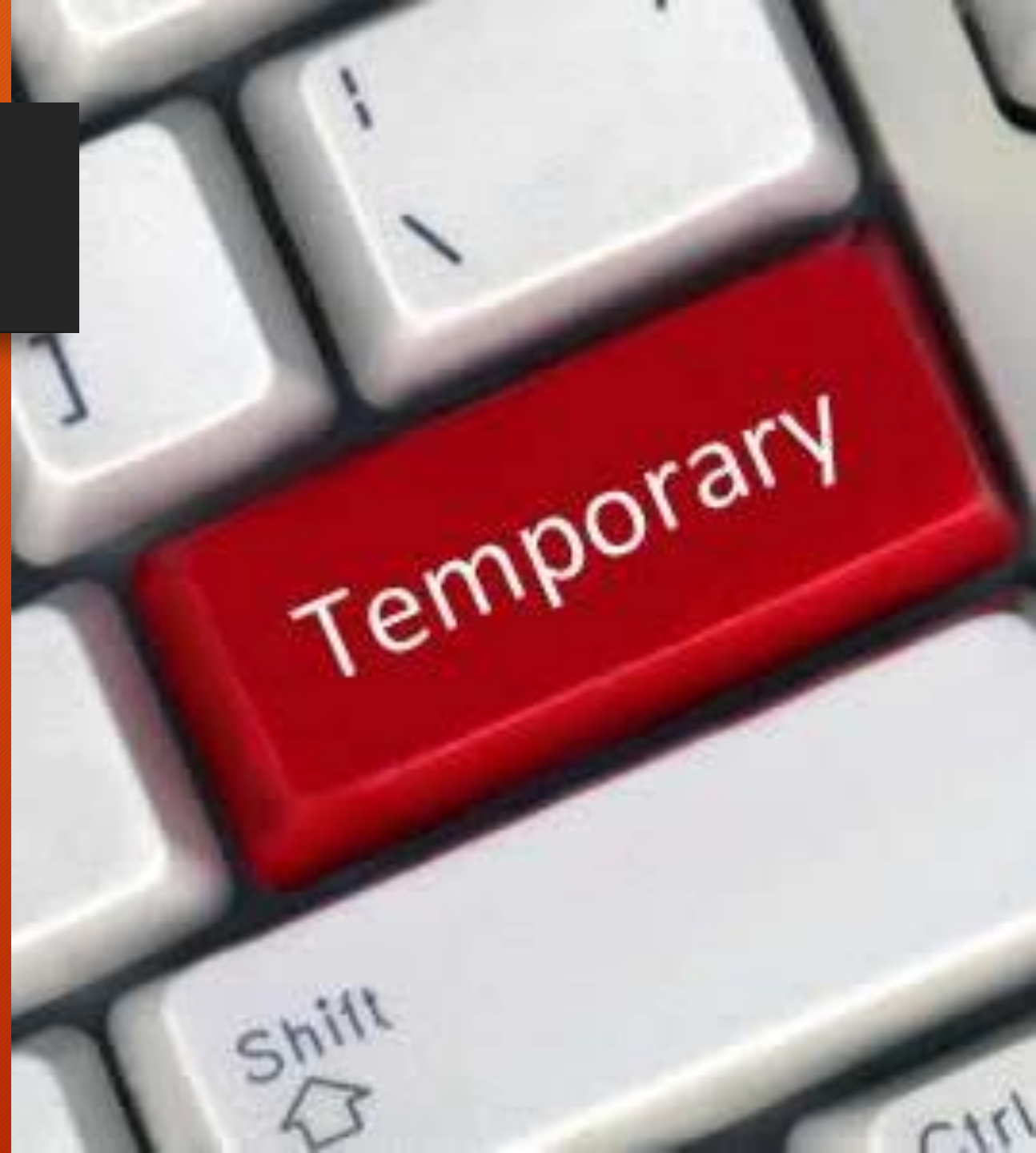
Toda query que resulta em uma grande quantidade de dados como resultado será armazenado em um Temporary Tablespace.



Tablespace temporário

Temporary Tablespace group

O **Temporary Tablespace Group** permite que um banco de dados Oracle escreva em múltiplos Temporary Tablespaces simultaneamente.

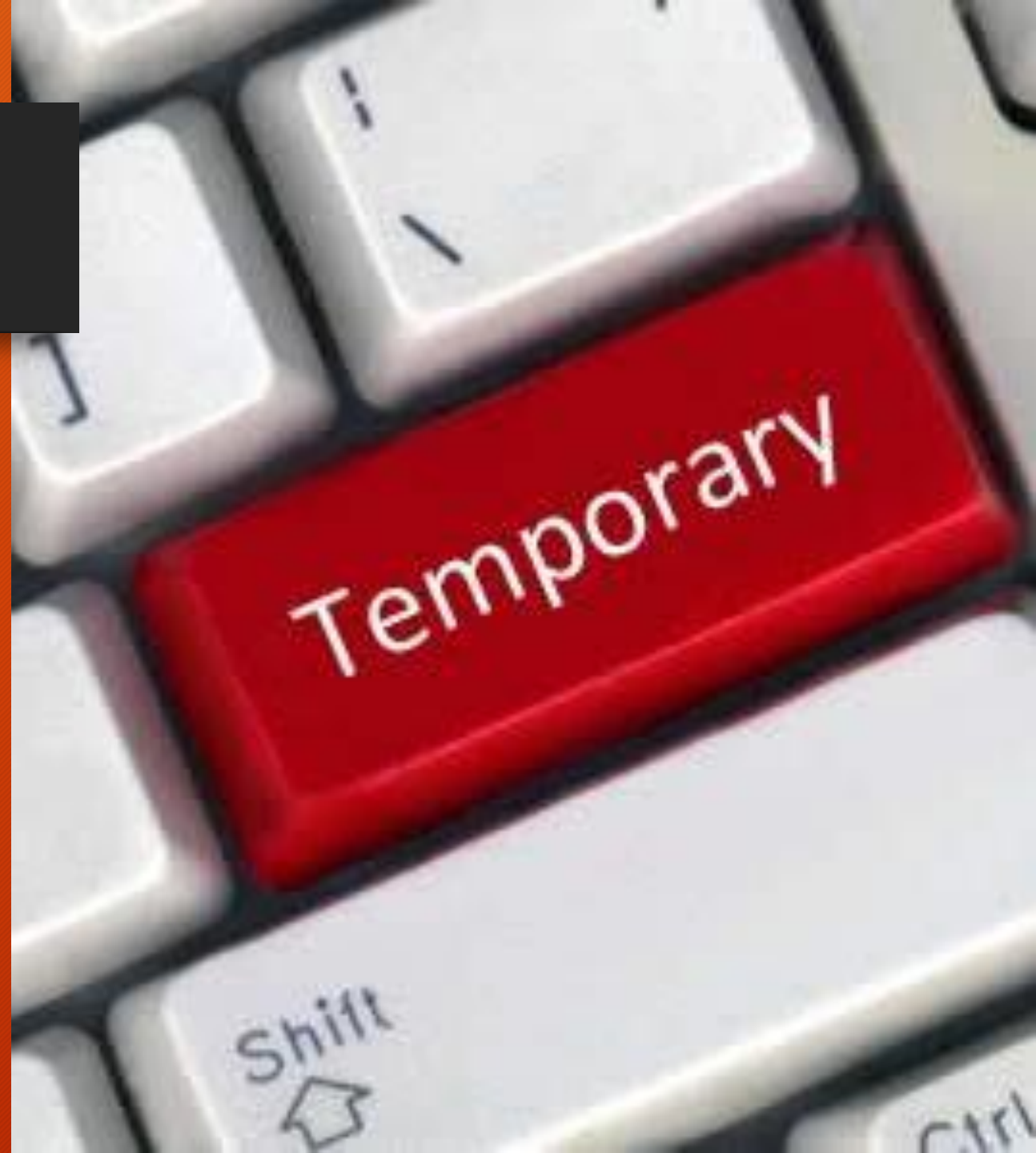


Tablespace temporário

Temporary Tablespace group

Um usuário fica atribuído a um grupo ao invés de um Temporary Tablespace simples;

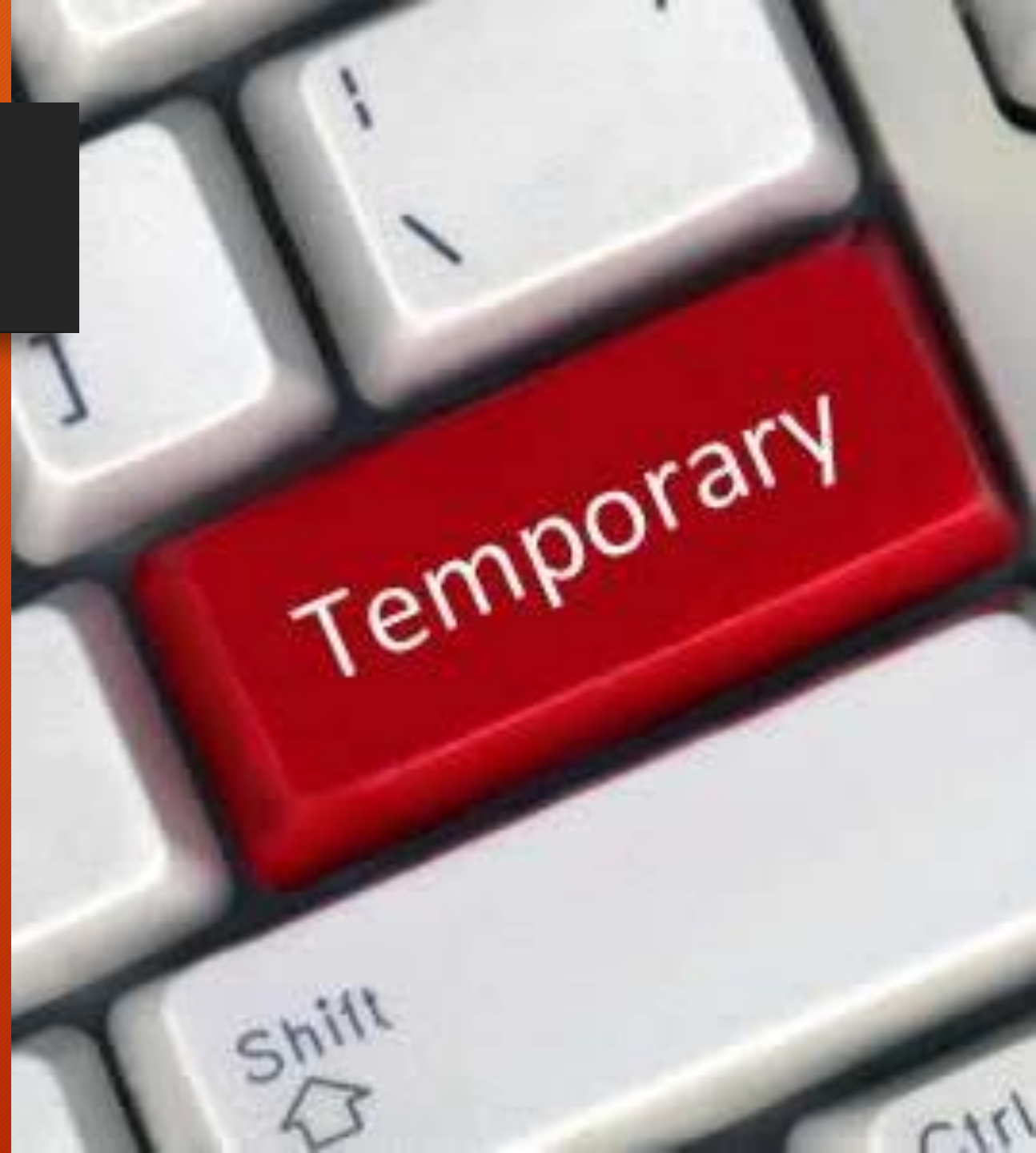
Uma única operação SQL pode usar múltiplos Temporary tablespaces.



Tablespace temporário

Temporary Tablespace group

A criação de um grupo ocorre no momento da criação de um Temporary Tablespace e a partir daí podemos adicionar outros Temporary Tablespace ao grupo



UNDO Management

UNDO Data é a cópia dos dados antes de serem modificados por uma transação que altera dados.



UNDO Management

UNDO Data é usado para:

- Transações de Rollback;
- Suportar leitura de consultas consistentes;
- Suportar operações de flashback;
- Recuperar-se de transações que falham.



UNDO Management

CUSTOMER_ID	Name	Country
10	JOHN	USA
11	TOM	USA

INSERT

```
INSERT INTO  
CUSTOMER (customer_id,  
name, country)  
values (12,'JOE','USA');
```

OP 5.1 (11.1)
Delete Row Piece - DRP
Slot 4:

UPDATE

```
UPDATE CUSTOMER  
SET  
name = 'JEFF',  
country='IRELAND'  
WHERE customer_id = 10;
```

OP 5.1 (11.1)
Update Row Piece - URP
Slot 4:
c1: JOHN
c2: USA

DELETE

```
DELETE FROM  
CUSTOMER  
WHERE  
customer_id = 11
```

OP 5.1 (11.1)
Insert Row Piece - IRP
Slot 4:
C0: 11
C1: TOM
C2: USA

Undo

UNDO Management

Os dados em UNDO permanecem no disco até:

- Usuário comitar (commit) a transação;
- Usuário reverter (rollback) as transação.



UNDO Management

Os dados em UNDO permanecem no disco até:

- Usuário execute algum comando DDL (CREATE, DROP, ALTER e RENAME) na tabela para o qual os dados são mantidos.



UNDO Management

Os dados em UNDO permanecem no disco até:

- A sessão do usuário é terminada de forma “anormal”, e neste caso ocorrerá um rollback e os dados em UNDO não serão mais mantidos.



UNDO Management

Os dados em UNDO permanecem no disco até:

- A sessão do usuário é terminada de forma “normal”, e neste caso ocorrerá um commit e os dados em UNDO não serão mais mantidos.



UNDO Management

Onde ficam os dados de UNDO:

- Armazenados em UNDO tablespace;
- Somente um tablespace ativo para uma instancia;
- Pertencem ao usuário SYS.



User Management

Uma conta de usuário do banco de dados é uma forma de **organizar** os proprietários dos objetos do banco de dados e os **direitos** de acesso.



User Management

Cada usuário do banco de dados possui:

- Um nome de usuário único;
- Um método de autenticação;
- Um tablespace default;
- Um tablespace temporário;
- Um user profile;
- Um status de conta.



User Management

O que é um Schema?

- É uma coleção de objetos de um banco de dados que pertencem a um usuário;
- Tem o mesmo nome que a conta do usuário.

Para saber mais:

[Understanding Database, Instance and Schema in Oracle database](#)



User Management

Contas administrativas padrão (default)

- **SYS**

- Usado pelo DBA;
- É o usuário master;
- Possui todos os privilégios;
- Pode executar operações de inicialização e parada do banco de dados;
- Pode efetuar backup e recovery do banco de dados;
- Pode fazer upgrade do banco de dados;



User Management

Contas administrativas padrão (default)

- **SYSTEM**

- É uma conta similar ao SYS;
- Não pode efetuar backup e recovery;
- Não pode efetuar upgrade do banco de dados;
- Deve ser utilizada quando não precisar das operações exclusivas do SYS.



User Management

Contas administrativas padrão (default)

- **DBSNMP**

- É uma conta utilizada pelo Oracle Enterprise Manager para monitorar e gerenciar o banco de dados;
- O Oracle Enterprise Manager é uma ferramenta gráfica utilizada para gerenciar o Oracle database.



User Management

Contas administrativas padrão (default)

- **SYSMAN**

- Também é utilizada pelo Oracle Enterprise Manager para realizar tarefas administrativas do banco de dados;
- Pode ser utilizado para várias tarefas do banco de dados, como criar tablespace e usuários;
- Também pode ser utilizado para efetuar backup e recovery.



User Management

Como criar um usuário?



```
create user pedro identified by 123;
```

```
create user pedro identified by 123 password expire;
```

User Management

Como criar um usuário?

```
create user pedro  
profile DEFAULT  
identified by 123  
default tablespace TBS1  
temporary tablespace TEMP  
account UNLOCK;
```

```
grant connect to pedro;  
grant <recurso> to pedro;
```



User Management

Trocar senha de um usuário?

```
alter user pedro  
identified by pedro123;
```



User Management

Privilégios

Privilégio é um direito de **executar** um tipo de instrução SQL ou **acessar** objetos de banco de dados de outro usuário. Há dois tipos de privilégios:

- System privileges
- Object privileges.



System
Privilege

Object
Privilege

User Management

System Privilege

Permite o usuário executar uma operação particular do banco de dados:

- CREATE TABLE;
- CREATE TABLESPACE;
- CREATE DATABASE LINK;
- CREATE SYNONYM;
- CREATE VIEW;
- CREATE SESSION;
- UNLIMITED TABLESPACE



Sintaxe:

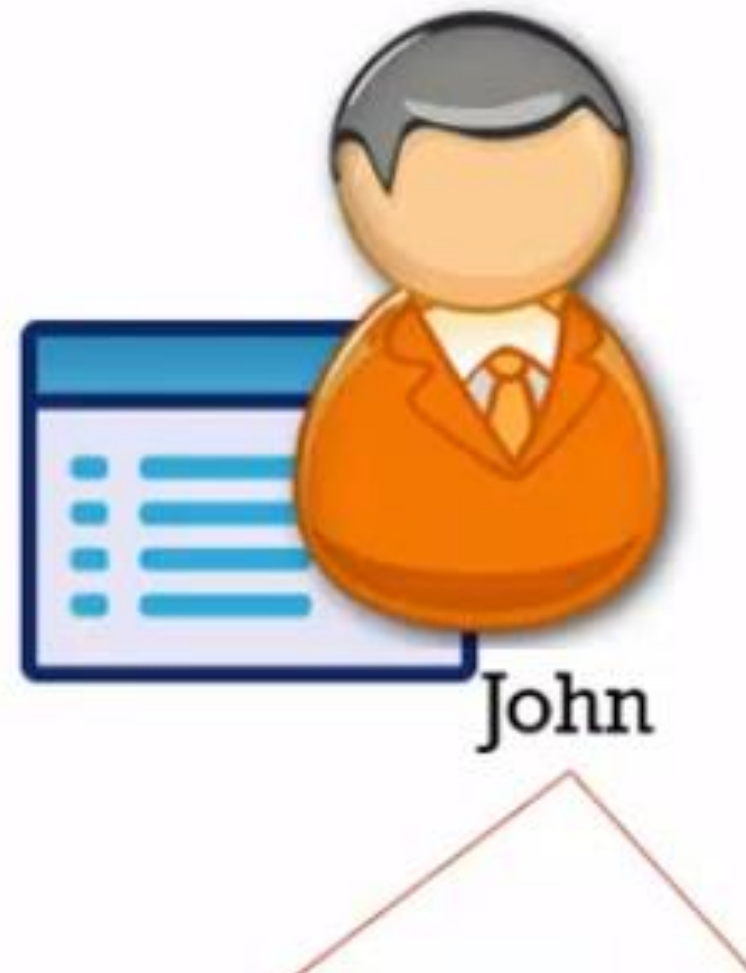
```
GRANT <system_privilege> TO <grantee clause> [WITH ADMIN OPTION]
```

User Management

System Privilege

ANY clause

Garante privilégios ao usuário em todos os Schemas:



Exemplo:

```
GRANT CREATE TABLE TO pedro;
```

Pode criar tabela somente no Schema do pedro

```
GRANT CREATE ANY TABLE TO pedro;
```

Pode criar tabela no Schema de todos os usuários

User Management

Object Privilege

Permite o usuário executar uma operação em um objeto específico, como uma tabela, view, sequence, procedure, function, etc.:

- SELECT
- UPDATE
- INSERT
- ALTER
- EXECUTE



Sintaxe:

```
GRANT <object_privilege> ON <object> TO <grantee clause> [WITH GRANT OPTION]
```

Exemplo:

```
GRANT DELETE ON pedro.cliente TO ana;
```

```
GRANT DELETE ON pedro.cliente TO ana WITH GRANT OPTION;
```

User Management

Object Privilege

Privilege	Description
SELECT	<i>select statement on tables</i>
INSERT	<i>insert statement on the table</i>
DELETE	<i>delete statement on the table</i>
INDEX	<i>Create an index on an existing table</i>
CREATE	<i>Create table statements</i>
ALTER	<i>Ability to perform ALTER TABLE to change the table definition</i>
DROP	<i>Drop table statements</i>
ALL	<i>Grant all permissions except GRANT OPTION</i>
UPDATE	<i>Update statements on the table</i>
GRANT	<i>Allows to grant the privilege that</i>

User Management

Revogação de privilégios

Revogar um privilégio significa **remover** os privilégios concedidos a um usuário.

Há diferenças no comportamento da revogação de Object Privileges e System Privileges.



System
Privilege

Object
Privilege

User Management

Revogação de privilégios - SYSTEM

GRANT



DBA

GRANT CREATE TABLE
TO pedro
WITH ADMIN OPTION;



PEDRO

GRANT CREATE
TABLE TO ana;



ANA

REVOKE



DBA

REVOKE CREATE TABLE
FROM pedro;



PEDRO



ANA

User Management

Revogação de privilégios - OBJECT

GRANT



DBA

GRANT SELECT ON
BOB.VENDAS TO pedro
WITH GRANT OPTION;



PEDRO

GRANT SELECT
ON BOB.VENDAS
TO ana;



ANA

REVOKE



DBA

REVOKE SELECT
ON BOB.VENDAS
FROM pedro;



PEDRO



ANA