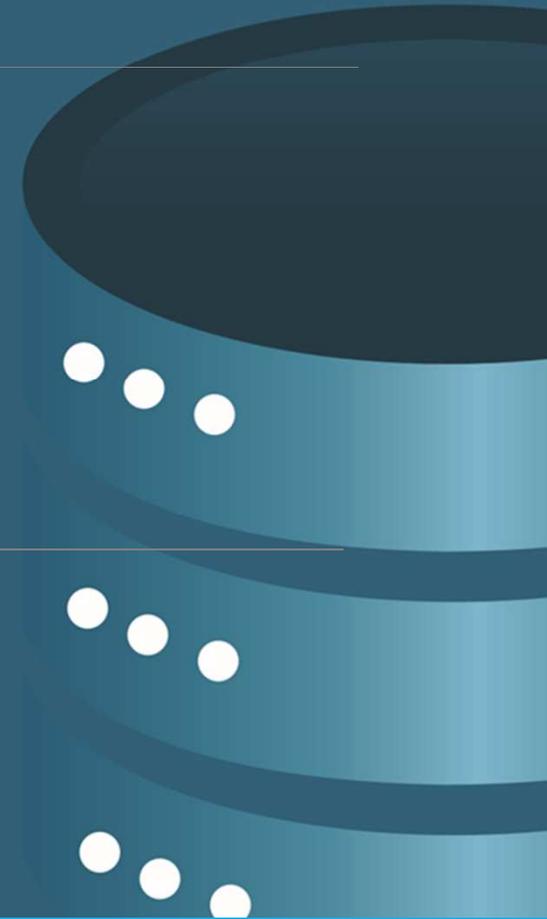


AULA 01

**BANCO DE DADOS: O QUE É?
PARA QUE SERVE? QUAIS EXISTEM??**



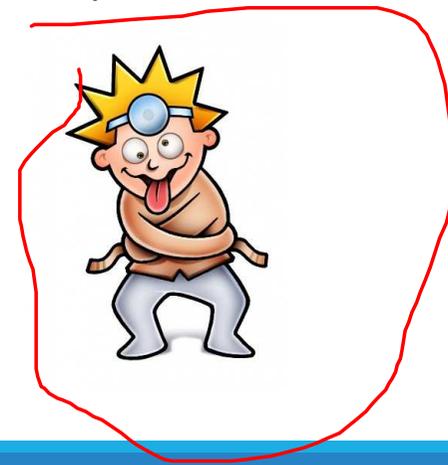
O que é um banco de dados??

Bem, antes de explicar vou te levar a uma reflexão importante.

Quais os nomes dos seus colegas de trabalho que sentam próximos a você?

Fácil né??? Vamos dificultar um pouco...

Quais os nomes dos funcionários que trabalham na mesma empresa que você, onde moram, quais seu emails?



CaLMA!!!

Isso foi apenas para te falar que bancos de dados surgiram da grande necessidade que nós seres humanos temos de armazenar informações, ou seja...

Armazenar...

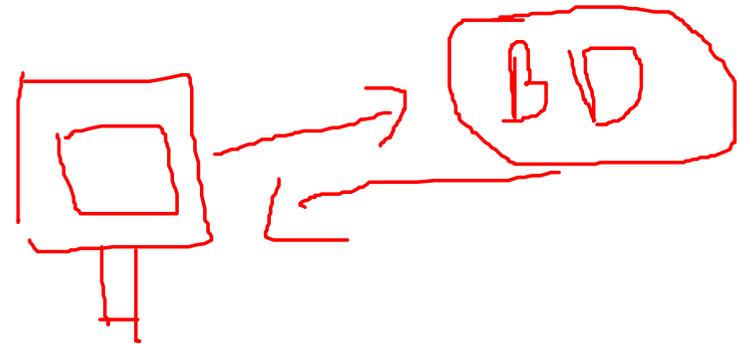


Banco de dados

Então nada mais é do que um local onde guardam-se dados, informações diversas.

Quais informações não interessa, porém o importante é que elas tenha alguns itens:

1. Sejam permanentes;
2. Acessíveis rapidamente (performance)
3. Segurança das informações (confidencialidade)
4. Fácil acesso.



Vejam bem, começamos a verificar alguns conceitos interessantes sobre bancos de dados.

Porém gerenciar os dados não é nada fácil. Controlar, manter, dispor...

Quem vai cuidar dessa parafernália toda
então??

SE VOCÊ RESPONDEU O DBA ADMINISTRADOR DE BANCO DE DADOS...

AVISO

Sinto muito EM INFORMAR... VOCÊ ERROU!!!



HUMM???

Errou! Mas mantenha a calma. Hehehe!!! Vou explicar onde está a falha no conceito.

Quem controla os dados, a forma como são disponibilizados, quem faz a entrega deles é algo chamado SGBD – Sistema Gerenciador de Banco de Dados, ou como temos em diversas literaturas em inglês: RDBMS – Relational Database Management System.

Ele é a ferramenta de controle e independentemente de qual fabricante seja sempre existirá um.

BANCO DE DADOS RELACIONAL

Bem como o próprio nome diz, RELACIONAL significa relacionamento. (meio óbvio né? Rsrtrs)

Na verdade vamos imaginar um exemplo onde tenhamos diversos dados a serem armazenados, várias informações.

Em um banco de dados chamado relacional, os dados serão armazenados em uma estrutura denominada tabela. Esta por sua vez é composta por linhas e colunas.

The diagram shows a table with two columns: 'Nome' and 'Email'. The first row contains 'José da Silva' and 'j.silva@teste.com'. The second row contains 'Rafael Crisaldo' and 'rafa.c@teste.com'. A red box highlights the entire table. A blue arrow points from the label 'linha' to the first row. Another blue arrow points from the label 'Coluna' to the 'Email' column.

Nome	Email
José da Silva	j.silva@teste.com
Rafael Crisaldo	rafa.c@teste.com

E por que relacional?

Então, Relacional significa que manterá certa relação.

Vamos imaginar que a tabela explicada anteriormente seja dos associados de um clube ou associação. Essa entidade então vai promover um curso de natação somente para os associados.

Teremos então uma tabela que irá registrar nome dos inscritos para o curso. Porém, existe uma condição para que possa fazer o curso lembra? Ser associado. Essa é a palavra chave para o relacionamento.

Então ninguém poderá estar na tabela de matriculados sem antes estar na tabela de associados.

Simple não é mesmo?? A este conceito damos o nome de integridade referencial e todo este processo será controlado pelo SGBD.

Um conceito muito importante:

Embora praticamente quase nunca seja comentado desta forma nos diversos treinamentos de banco de dados que ministrei existe um conceito muito interessante que preciso compartilhar com vocês.

Ele se chama: A.C.I.D.

A -> Atomicidade: Um conjunto de transações efetuado deve ser feito por completo, não haverá fazer uma transação parcialmente.

C -> Consistência: mantém a consistência de dados

I - > Isolamento: As transações são isoladas umas das outras antes de serem confirmadas.

D- > Durabilidade: Após confirmadas as alterações feitas por uma transação ela permanecerá permanentemente (ou até que alguém a altere novamente é claro).

Tecnologias de banco de dados

Como comentado a pouco, já sabemos que precisamos armazenar dados, que eles tem que ser garantidos e que tudo isso é controlado por um SGBD, certo?

No mercado hoje temos diversos sistemas aptos para trabalharem em grandes ambientes com boa disponibilidade.

DENTRE OS MAIS UTILIZADOS HOJE TEMOS:



O melhor entre os melhores

Embora tenhamos diversos sistemas, cada qual com seu nível de performance, segurança, custo, facilidade, etc sempre existirá aquele com qual o “negócio” (Business) mais se adapta.

É claro que precisariam ainda ser analisadas questões de custos, mão de obra especializada, etc. Muitos são os itens que influenciam na escolha.

Eu particularmente conheço bem Oracle Database e MySQL sendo que o primeiro tem me acompanhado pelo últimos 16 anos da minha carreira e sendo ele o foco deste treinamento.

Portando se você DBA, estudante, curioso, está aí, fazendo um curso Oracle DBA e iniciando no mundo de banco de dados já por esta maravilhosa ferramenta, eu só tenho uma palavra a te dizer...

BEM VINDO AO MARAVILHOSO MUNDO

PARABÉNS!!!



Um pouco da evolução

Como a maioria dos softwares do mercado o Oracle evoluiu muito desde sua criação em 1978 por Larry Ellison que é o presidente da empresa até hoje.



i – Internet
g – Grid
c – Cloud

Administrar um Banco de Dados

Pois bem... Para que tudo isso dê certo, ou seja, para que um banco seja gerenciado pelo SGBD e este seja configurado administrado de forma correta precisamos de uma figura muito especial.



Ops... Ele não! Me enganei... Agora sim!



DBA

O que é atribuição do DBA??



Essa é uma pergunta muito interessante e pertinente, e respondo ela com uma afirmação: **DEPENDE!**

Sim, uma vez que a carreira DBA sempre foi uma das que prezou muito pela divisão Jr, Pleno Sênior. As atribuições irão aumentando de acordo com o nível de experiência.

Este primeiro módulo do nosso curso é voltado ao DBA iniciante, ou seja, DBA Jr.

Porém é fato que DBAs pleno, e Sênior precisarão ter conhecimento total disso. Então todos os níveis irão se beneficiar.

DBA Jr.

Acompanhando as tendências do mercado que cada vez tem sido mais exigente, coloquei no curso alguns atributos que sinceramente eu acho que seriam mais para DBA Pleno que Jr, mas que poderá ajudar e muito, principalmente para quem está no mercado de trabalho ou querendo entrar nele.

Aqui iremos desde o início do processo, passando pela instalação, upgrade de versão, configurações e gerenciamentos.

Então vamos lá! Desvendar essa caixa preta chamada ORACLE.

A Estrutura do Servidor Oracle

Um sistema SGBD para que ele possa cumprir suas atribuições de gerenciar dados, manter integridade e segurança necessita de uma estrutura física.

Esta estrutura chamamos de **servidor de banco de dados**, e é composto por:

Servidor físico ou virtual – Máquina onde será instalado e configurado;

Sistema operacional;

Conexão de rede;

Sistema de armazenamento (storage) .

Servidor Físico ou Virtual

Chamamos servidor a máquina física ou virtual (veremos o conceito de virtual mais adiante) que contém um Sistema operacional onde será instalado o banco de dados.

Nesse momento muitos pensam *“Qual a melhor máquina para se instalar um banco de dados Oracle?”*

A resposta clássica de sempre: **DEPENDE!** Depende da necessidade do negócio que o banco de dados irá fazer parte.

Porém é fato que sempre teremos que ter um servidor onde instalar toda a nossa parafernália e com um mínimo de performance para o cliente final (usuário).

Deste servidor temos que saber que seus recursos de hardware serão muito importantes para todo o processo. Como veremos a seguir, pois um servidor Oracle é composto de basicamente dois itens.

A estrutura Oracle no servidor

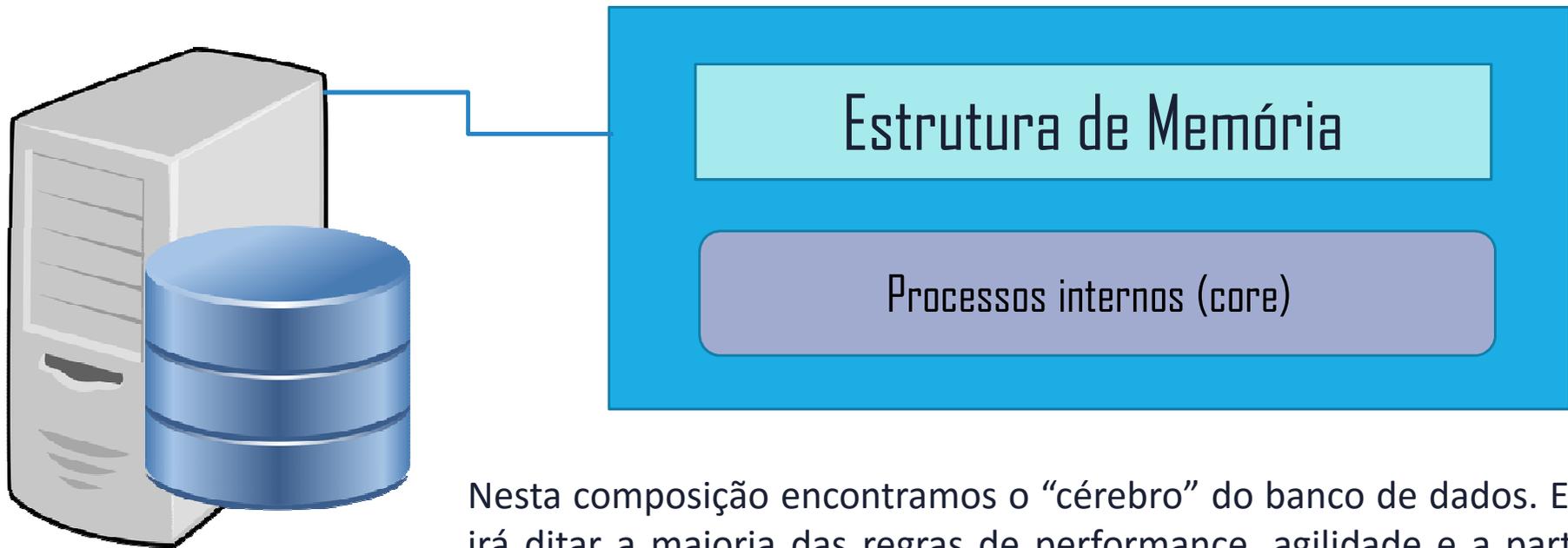


Instância

+

Armazenamento

Instância Oracle



Nesta composição encontramos o “cérebro” do banco de dados. Ela irá ditar a maioria das regras de performance, agilidade e a partir daqui começa o desvendar da caixa preta do Oracle.

Instância Oracle Estrutura de Memória

Quando falamos na estrutura de memória da instância Oracle a primeira coisa que devemos sempre lembrar é que ela está imbutida em um servidor que é um hardware (físico ou virtual) e que possui uma quantidade específica de memória RAM.

Sendo assim algumas coisas precisamos considerar:

- 1- A quantidade de memória é finita;
- 2- O sistema operacional também precisa dela pra trabalhar ela é **compartilhada**;
- 3- Por ser memória RAM, ela é volátil, ou seja, armazena dados enquanto o servidor está ligado.
- 4- Nem sempre quanto mais memória melhor (guarde esse conceito, um dia falaremos mais sobre ele)

Instância Oracle Estrutura de Memória

Essa memória RAM conforme falado anteriormente é compartilhada no servidor pelo sistema operacional e pela instância Oracle de forma a trabalharem juntos.

Portanto saber quanto de memória RAM definir para a instância Oracle é algo primordial para que se tenha uma performance da estrutura Oracle boa porém sem impactar o sistema operacional.

Quando você define um valor de memória para a sua instância de banco de dados Oracle, ela é bem “radical” e quando o banco de dados “sobe” ele já aloca para ele toda a memória que foi designada. Por isso **CUIDADO!!!**

Saber quanto de memória alocar é primordial.

Mas qual seria esse valor tido como ideal???

Pelos padrões de melhores práticas este valor é em torno de 40% do total da memória RAM, no caso de termos apenas uma instância de banco de dados no servidor é claro.

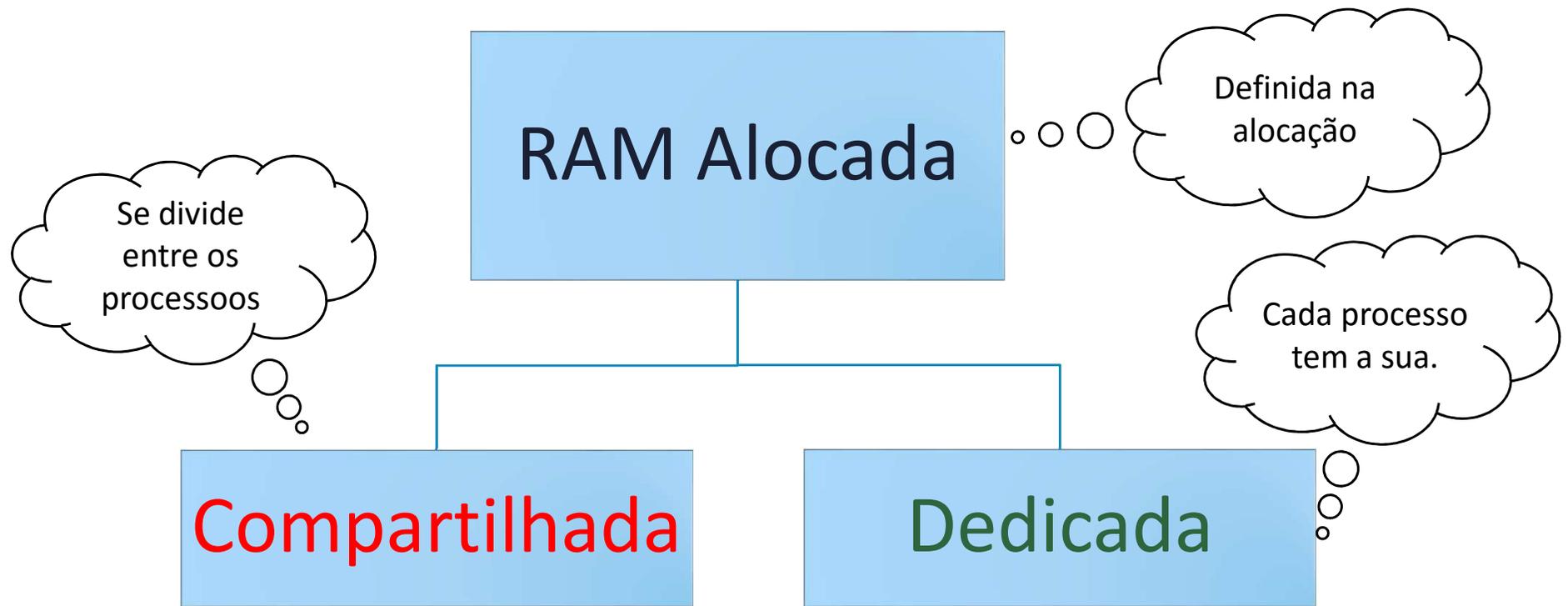
Porém pela minha prática, o meu “melhores práticas” é 60%.

Instância Oracle Estrutura de Memória

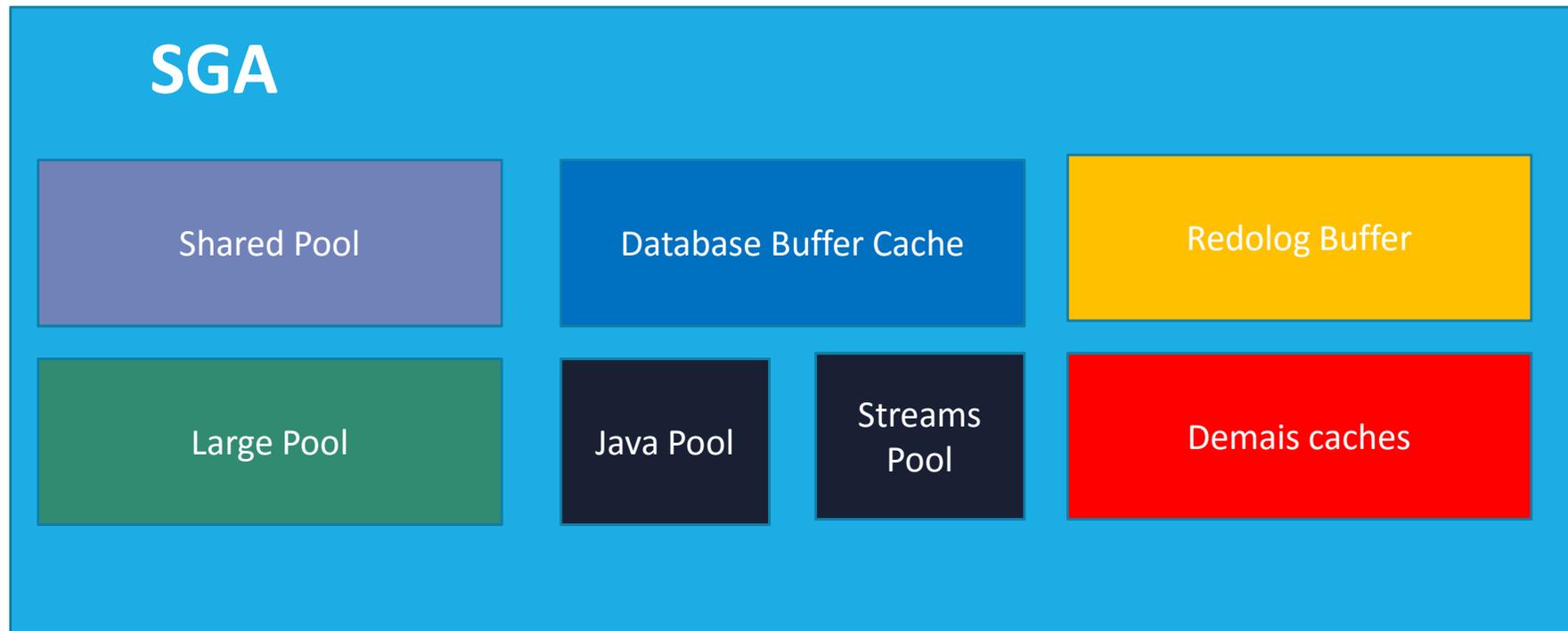


Como o Oracle utiliza esta memória???

Estrutura de Memória - Utilização



Estrutura de Memória - Compartilhada



Estrutura de Memória - Compartilhada

SGA

System Global Area, ou ainda Shared Global Area, é o nome dado a esta estrutura. Como o próprio nome sugere, trata-se de uma memória que é compartilhada por todos os processos que estiverem trabalhando no banco de dados.

Sem dúvida esta é uma das maiores “sacadas” do banco de dados, que vai permitir que diversos processos compartilhem tarefas. Assim se evitamos grande quantidade de retrabalho.

Para ficar mais fácil o entendimento vamos verificar cada área de memória das citadas anteriormente.

Compartilhada – SHARED POOL

Shared POOL é uma área de memória muito importante, pois ela é o início de todo o processo de busca de dados.

Na verdade o que é um sistema de banco de dados senão um sistema onde alguém faz a pergunta **Aplicação, client, usuário**; esta é feita na linguagem que o banco de dados entende, no nosso caso o SQL, e ele responde, trazendo na tela um resultado. Simples assim...

Porém para que isso ocorra da forma realmente como se espera diversos processos estão envolvidos.

E como eu disse: Tudo começa na Shared pool.

Quando uma requisição é enviada para o banco de dados, ou seja, quando um select é solicitado, quem recebe esta informação é a shared pool.

Lá irá ocorrer uma fase muito importante e estudada em processo de tuning de banco de dados (performance) chamada **PARSE**.

Ela recebe o SQL e verifica: **Sintaxe, existência objeto solicitado, permissão para acessar**.

Após acessado ela verifica qual a melhor forma de trazer os dados e nesse momento ela monta o **plano de execução**.

Agora começa uma das atribuições mais importantes desta área.

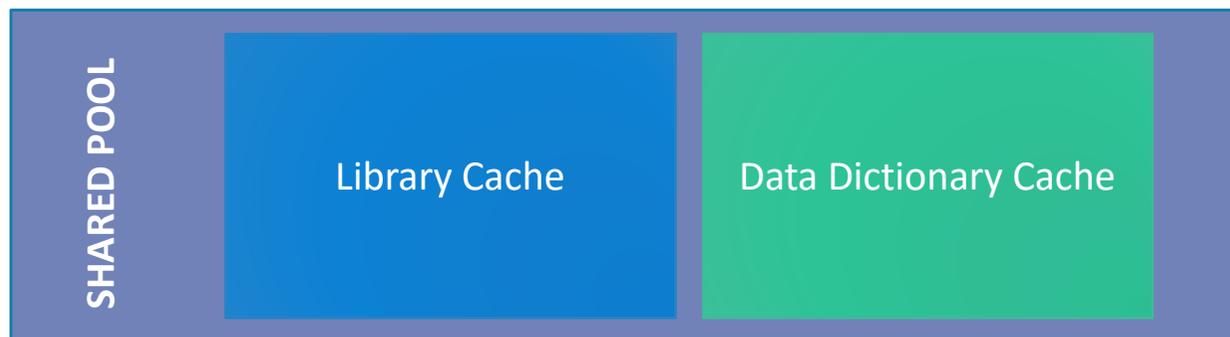
Compartilhada – SHARED POOL

Ela armazena este plano que é uma “receita de bolo” de como acessar os dados. Com ele armazenado e catalogado, qualquer outra sessão que fizer a mesma pergunta pro banco (SELECT) ele já saberá como trazer os dados e isso não mais será um parse completo e sim muito mais rápido.

Outra função muito importante deste pool de memória é armazenar o dicionário de dados

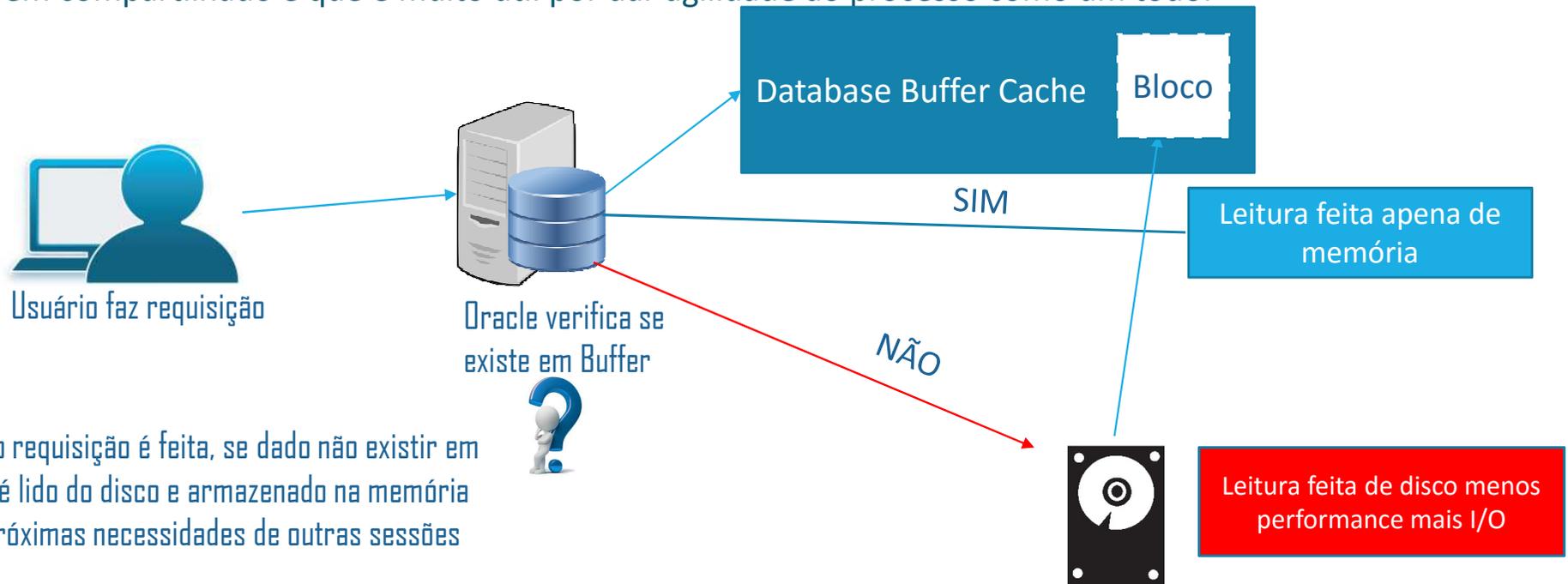
Este dicionário são informações importantes do banco de dados que são armazenados em memória e que poderão ser lidos mesmo com o banco de dados ainda “não aberto” e muito úteis em algumas situações de manutenção.

Veja abaixo como é subdividida a **SHARED POOL**



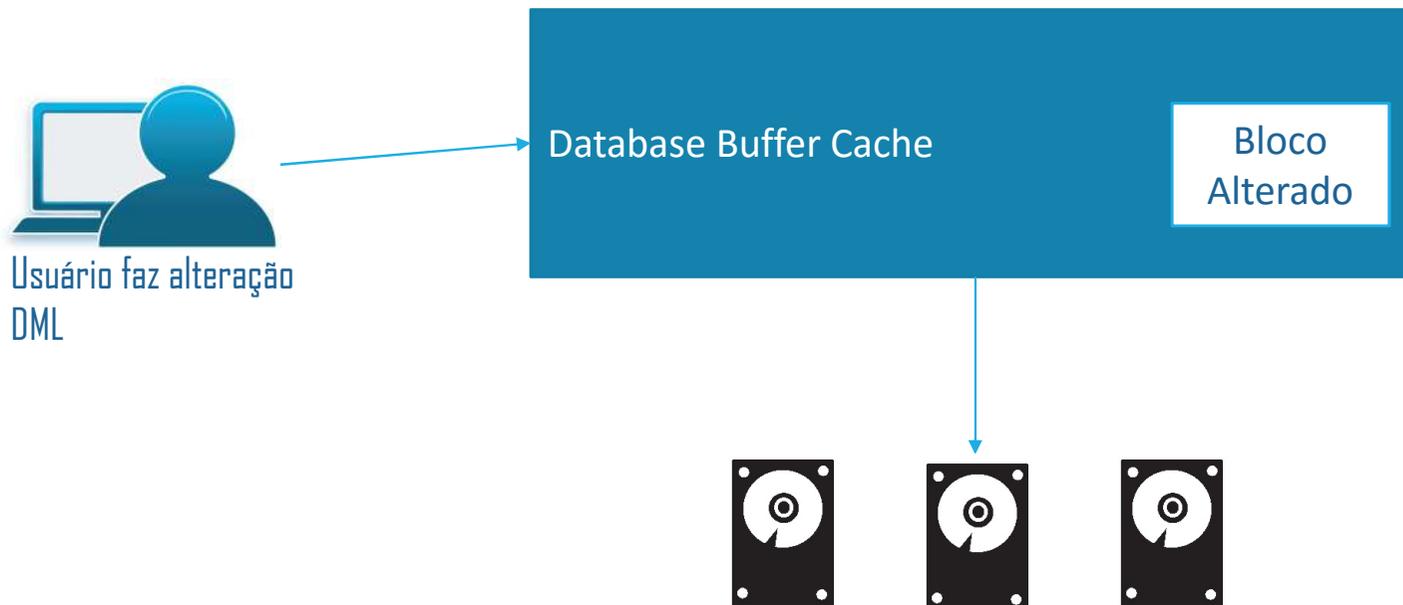
Compartilhada – Database Buffer Cache

Database Buffer Cache é um buffer de memória, ou seja, um local de armazenamento de memória, também compartilhado e que é muito útil por dar agilidade ao processo como um todo.



Database Buffer Cache – Gravando dados

Database Buffer Cache também tem a função de armazenar as alterações feitas pelos usuários e que depois se tornarão definitivas no momento da confirmação (COMMIT). Veremos o processo mais adiante.

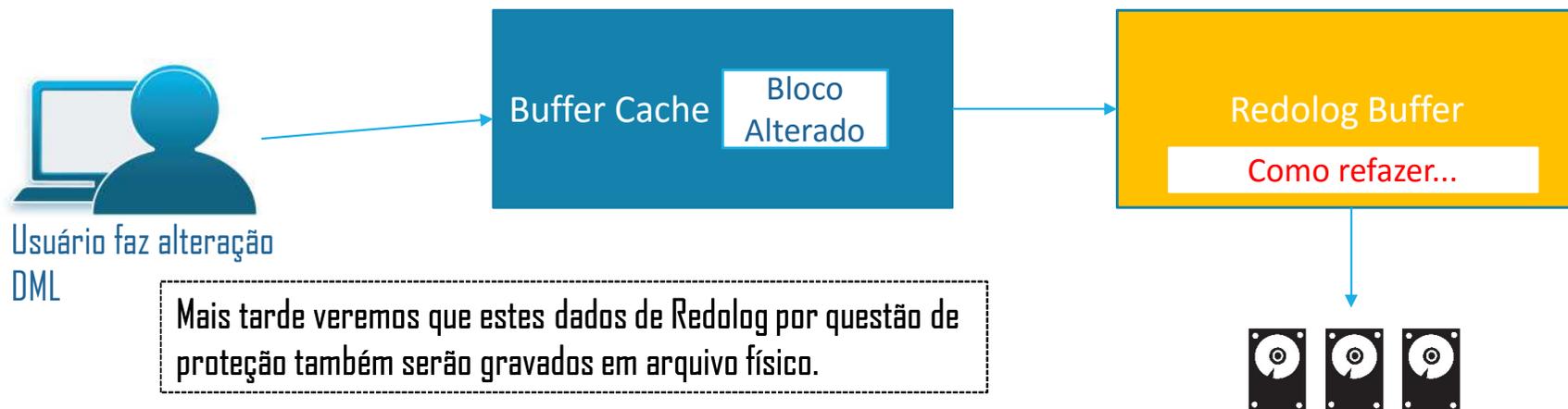


Compartilhada – REDOLOG BUFFER

Redolog Buffer é juntamente com os demais um local de armazenamento muito importante, como o próprio nome diz, **RE-DO = RE-FAZER**, ou seja, nos diz como refazer alguma tarefa caso seja necessário.

Como Funciona???

Imagine que uma alteração seja feita em algum dado, exemplo, um update. Esta alteração será processada nos blocos que foram armazenados no buffer conforme vimos anteriormente e o redolog terá armazenada a informação de como refazer isso caso seja necessário.



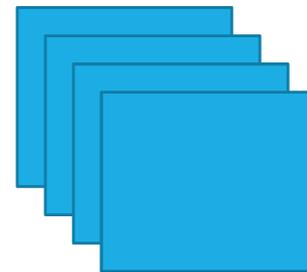
Compartilhada – LARGE POOL

Large Pool: É uma porção também da SGA que tem a função de agilizar alguns processos que utilizam bastante processamento.

Exemplos principais:



Backup/Restore



**Operações com
paralelismo**

Sendo a Large Pool mal dimensionada teremos problemas pois será ocupada área da Shared Pool.

Estrutura de Memória - Dedicada

PGA

Program Global Área, é uma área de memória que é dedicada exclusivamente a cada processo criado no servidor.

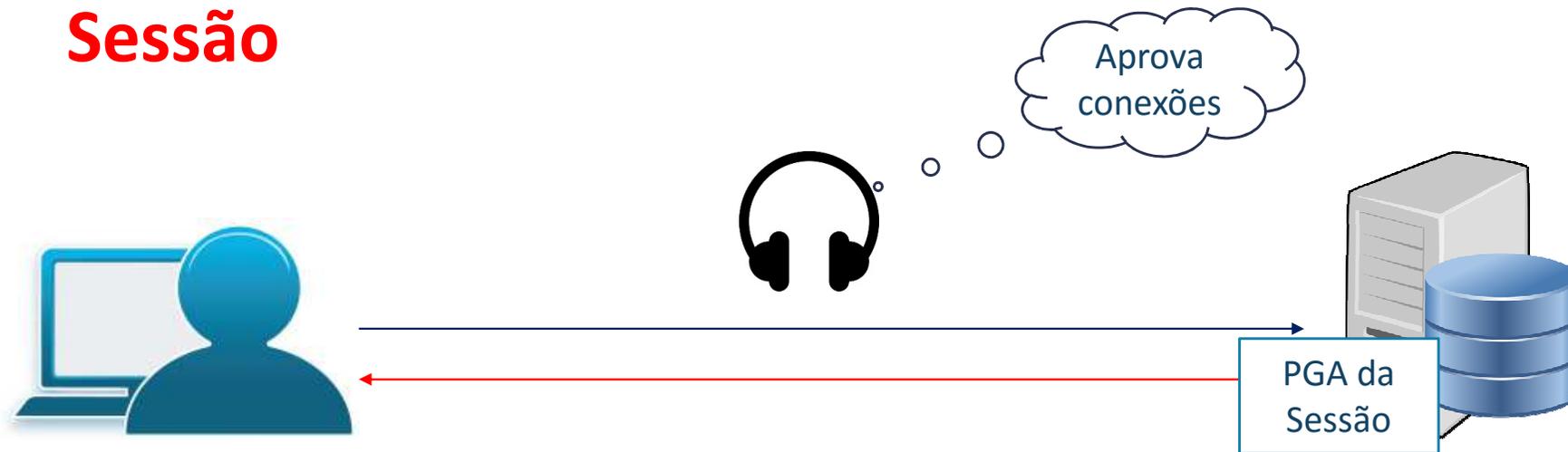
Ou seja, ela é única de cada sessão.

Mas o que é sessão???



Dedicada – PGA

Sessão



Toda vez que um usuário precisa recuperar informações de um banco de dados ele o faz através da solicitação de conexão. Esta é feita via rede utilizando-se normalmente um client do Oracle.

Esta solicitação de conexão é recebida por um objeto que faz parte da estrutura que se chama **Listener**.

Caso seja **aprovada** a conexão, automaticamente é criada uma sessão no banco de dados com uma **PGA** específica pra ela e o Listener não tem mais nada a ver com o processo.

Instância Oracle – Processos Internos

Sempre que um banco de dados é iniciado, existem alguns processos que são iniciados juntamente com ele no sistema Operacional. A este conjunto de processos damos o nome de Processos Core ou como mais popularmente são conhecidos: **Processo de Background**.

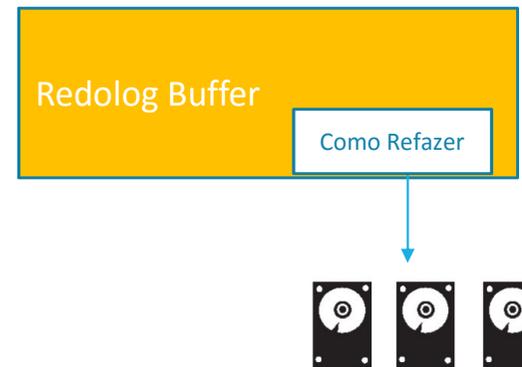
Já comentamos superficialmente sobre alguns deles em slides anteriores e iremos agora conhecê-los mais a fundo bem como alguns outros processo também importantes.



Processos Internos – LOG WRITER(LGWR)

LOG WRITER
LGWR

Como os buffers são todos dados em memória e visando garantir caso tenhamos um problema de “cair” o banco de repente, os dados devem ser gravados periodicamente para o disco e quem faz isso é o LOG WRITER.



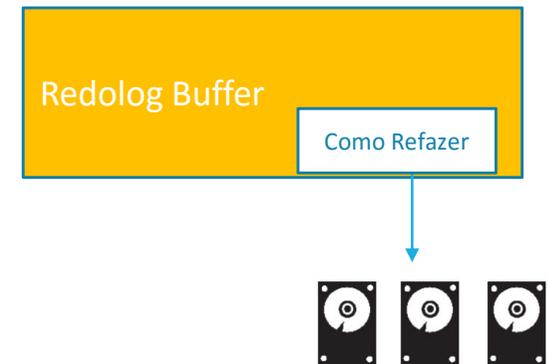
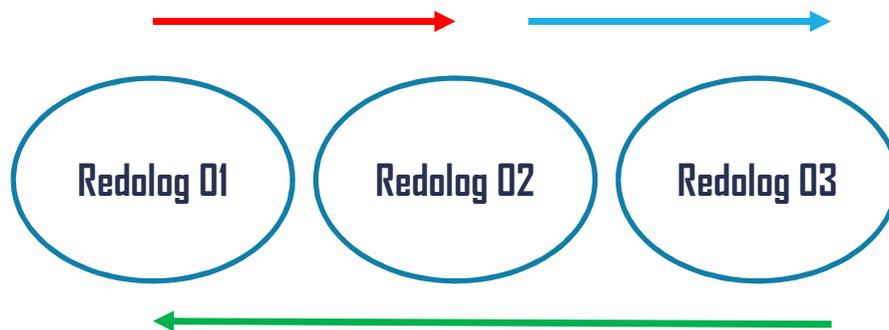
Em qual momento ele procede??

- 1- Quando um usuário processa um commit;
- 2- Quando o buffer está 1/3 cheio;
- 3- Antes do DBWn fazer a sua gravação;
- 4- A cada 3 segundos;

LOG WRITER(LGWR) – A gravação

A gravação é feita em arquivos físicos denominados **redolog online**. Estes arquivos tem tamanho fixo, definidos na instalação, podendo ser alterados futuramente.

Sua gravação é um processo cíclico, onde grava-se um arquivo, terminado o seu tamanho passa-se ao próximo e assim sucessivamente.

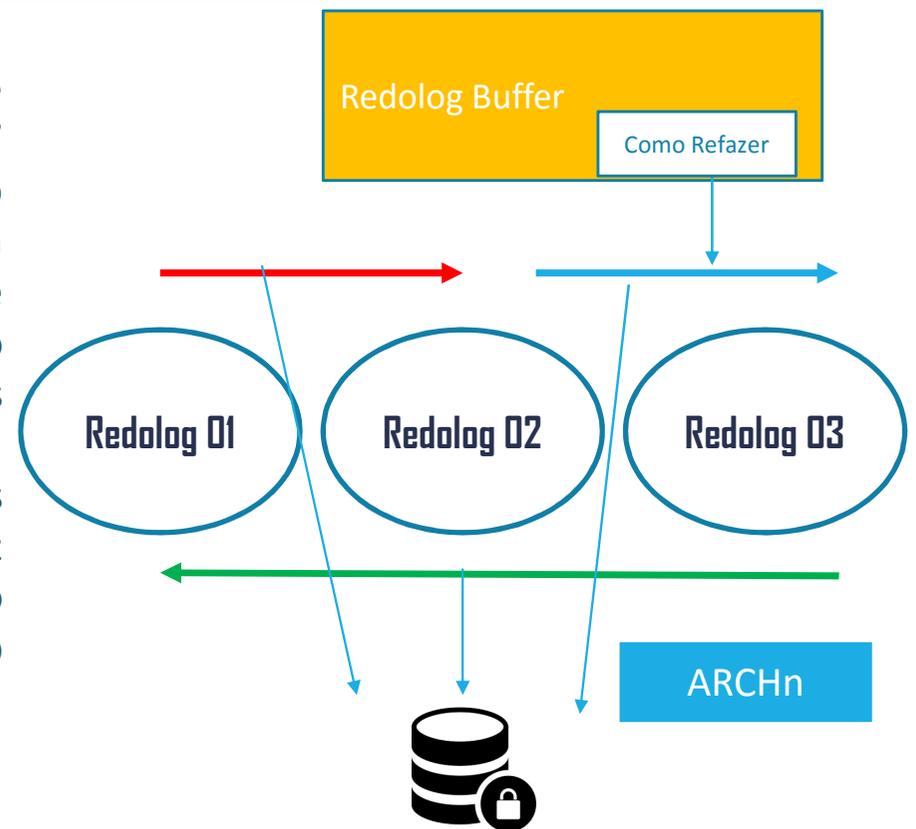


Assim cada vez que um arquivo é solicitado para ser gravado nele, o mesmo será limpo para receber novos dados.

Processos Internos – ARCHn

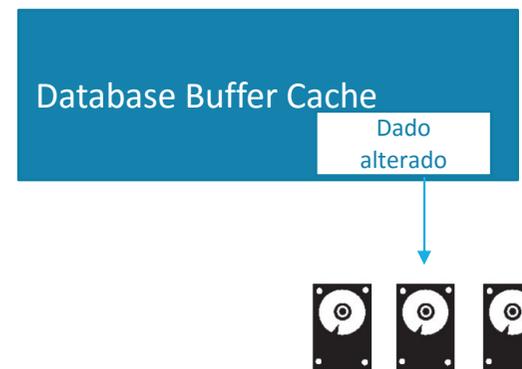
Conforme falado anteriormente a gravação do LGWR é feita de forma cíclica, ou seja cada vez que um redolog online é “chamado” para gravar dados ele é limpo, mas imagine se este dado ainda não tiver sido gravado no arquivo de dados final e tivermos um problema no banco de dados, afinal de contas o redolog online grava dados comitados e não comitados. Sendo assim corre-se o risco de não terem sido gravados em disco, mas serem necessários em caso de recuperação da instância pós problema.

Para resolver este problema pode-se configurar o banco de dados para trabalhar em modo **archivelog**, que é o processo que faz gravação dos dados do redolog online para um arquivo físico denominado **archivelog** este fica armazenado, inclusive em backup que pode ser utilizado quando necessário



Processos Internos – DB WRITER (DBWn)

DB WRITER
DBWn



Grava os blocos modificados chamados "dirty" para o disco, com isso garante que as alterações feitas por operações DML do cliente não sejam perdidas. Ocorre sempre após um processo de checkpoint e sempre é acompanhado de I/O de disco, ou seja, é gravação de disco fisicamente.

Ele é o processo responsável pelo avanço do Checkpoint que é o próximo processo que veremos a seguir.

Processos Internos – Check Point(CKPT)



Todo banco de dados deve ter seus dados íntegros como já vimos anteriormente, para isso se faz necessário um perfeito controle de versão dos arquivos físicos, para garantir que nem um esteja mais “novo” ou mais “velho” que o outro.

A forma como isso ocorre é logo após o processo que vimos do DB Writer existe um processo que “bate um carimbo” nos arquivos de banco de dados onde estão os dados **datafiles** e nos arquivos de controle **Controlfiles** trazendo a versão do banco de dados, controlada através de um número que será único após cada transação bem sucedida chamado de SCN – System Change Number. Assim quando o banco precisa ser recuperado ele irá procurar a versão por este número.

Processos Internos – Process Monitor(PMON)

Process
Monitor
PMON

Processo responsável por fazer a limpeza quando ocorre falha em algum processo de usuário.

- 1- Ele limpa o buffer cache específico que estava sendo utilizado pelo processo, exemplo uma DML que ainda não tinha sido dado commit;
- 2- Limpa os recursos que estavam sendo utilizados pelo processo, exemplo: alocação de memória PGA;
- 3- Monitora sessão caso esteja configurado verificar timeout de sessão idle;
- 4- Registra serviços do banco de dados no Listener (veremos mais adiante).

É um processo chamado essencial, caso você “mate” o processo no sistema operacional o banco será derrubado.

Processos Internos – System Monitor(SMON)

System Monitor SMON

Processo responsável por fazer o recovery da instância caso necessário.

Foi comentado a alguns slides atrás que o banco de dados deve sempre permanecer íntegro você se lembra?

Então, sempre que o banco cair de forma abrupta, ou até mesmo for baixado, sempre será verificado a necessidade de se proceder um Recovery. Este é o processo que equaliza o banco de dados com todos os SCNs iguais.

Caso seja necessária correção, a mesma será feita automaticamente por um processo chamado **SMON**.

Ele também é responsável pela limpeza dos dados temporários de uma sessão e que não precisem mais ser utilizados.

É um processo chamado essencial, caso você “mate” o processo no sistema operacional o banco será derrubado.

Preparativos para aula 02

Na próxima aula iremos iniciar a criação do ambiente virtual que será base do nosso curso e do seu treinamento extra-classe.

Precisaremos baixar:

1- VirtualBox Oracle para virtualização:

<http://www.virtualbox.org>

2- Versão do Oracle 11.2.0.3

<http://www.rauldba.com.br/databasell203.rar>

3- Oracle Linux

<http://www.rauldba.com.br/OrclLinux.zip>

Dicas da prova OCA 11g – 1z0-052

Conforme prometido, mesmo que para muitos sejam ainda estranhos os nomes, comandos, itens citados aqui, segue uma dica da prova de certificação.

Hoje a dica é para a prova **1z0-052 (OCA 11g)**. As dicas serão na grande maioria das vezes em inglês para que os que pretendem fazer a certificação já se familiarizem com o método de aplicação das provas da Oracle.

1- You notice that the performance of the database has degraded of frequent checkpoints. Which two actions resolve issue?

a) Disable automatic Checkpoint tuning

b) Check the size of the redolog file size and increase the size if it is small

c) Set FAST_START_MTTR_TARGET parameter as per th advice given by the MTTR Advisor

d) Decrease the number of redolog members if there are more than one redolog members available in each redolog group

Resposta:

Resposta correta:

b) Check the size of the redolog file size and increase the size if it is small

Primeiramente teremos que analisar o que faz ocorrerem os checkpoints. Como vimos, eles ocorrem pela troca de redolog ou processo de switch logfile que é aquele processo que faz a gravação que vimos que é cíclica. Ora, se ele ocorre checkpoints a cada vez que termina de gravar um redolog e passa para outro, logo se o tamanho do redolog for maior teremos menos trocas (switches) e conseqüentemente menos frequência de checkpoints.

c) Set FAST_START_MTTR_TARGET parameter as per th advice given by the MTTR Advisor.

O Oracle possui diversas ferramentas chamadas Advidores, a função deles normalmente é nos dar algumas dicas de configuração, performance, etc. Existe um que trata de itens voltados a performance do processo de recovery, lembrem do SMON que é responsável pelo processo de equalização do banco? Existe um advisor que nos dá dicas inclusive do tamanho ideal para o arquivo de redolog.