

Oracle Fundamentals



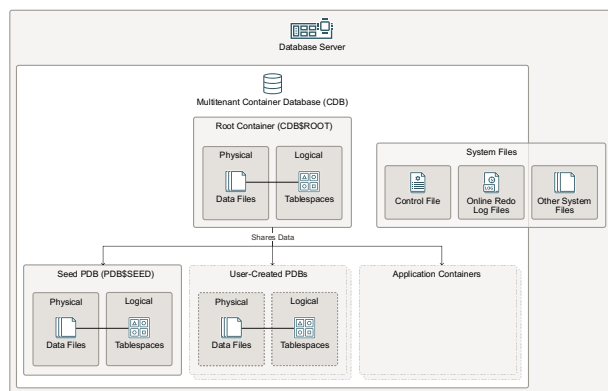
Database

SUMÁRIO

1.	DATABASE.....	4
1.1	BANCO DE DADOS "NON-CDB"	4
1.2	CONTAINER DATABASE (CDB).....	5
1.3	VANTAGENS DA ARQUITETURA MULTITENANT DATABASE.....	5
1.4	EXTRA OPTION	5
2.	MULTITENANT CONTAINER DATABASE.....	6
2.1	VANTAGENS.....	7
2.2	USO.....	7
2.3	ESTRUTURA.....	8
2.4	REFERÊNCIAS.....	8
3.	ESTRUTURAS DE ARMAZENAMENTO	9
3.1	ESTRUTURA DE ARMAZENAMENTO FÍSICA.....	9
3.2	ESTRUTURA DE ARMAZENAMENTO LÓGICA.....	9
4.	ESTRUTURA DE ARMAZENAMENTO FÍSICA	11
4.1	ARQUIVOS DE DADOS.....	11
4.2	ARQUIVOS TEMPORÁRIOS.....	12
4.3	ARQUIVOS DE CONTROLE	12
4.3.1	MULTIPLEXAÇÃO.....	13
4.4	REDO LOG.....	14
4.4.1	MULTIPLEXAÇÃO.....	15
4.4.2	REDO LOG SWITCH	15
5.	DATABASE SYSTEM FILES.....	16
5.1	ARQUIVOS DE CONTROLE	16
5.2	PARAMETER FILE.....	16
5.3	ARCHIVE LOG.....	17
5.3.1	MODO ARCHIVELOG	17
5.4	AUTOMATIC DIAGNOSTIC REPOSITORY (ADR).....	18
5.5	PASSWORD FILE	19
5.6	BLOCK CHANGE TRACK FILE	20

5.7	FLASHBACK DATABASE	20
5.8	REFERÊNCIAS	21
6.	LOCAIS DE ARMAZENAMENTO DOS DATABASE FILES	22
6.1.1	FILE SYSTEM	22
6.1.2	ASM	22
6.2	ORACLE MANAGED FILES	23
7.	ESTRUTURA LÓGICA DO BANCO DE DADOS	24
7.1	REFERÊNCIAS	24
8.	TABLESPACES	25
8.1	SYSTEM TABLESPACE	25
8.2	SYSAUX TABLESPACE	25
8.3	UNDO TABLESPACE	26
8.4	USER TABLESPACE	26
8.5	TEMPORARY TABLESPACE	27
8.6	BIGFILE TABLESPACE	27
8.7	SMALLFILE TABLESPACE	28
8.8	ARQUIVOS FÍSICOS	28
8.9	REFERÊNCIAS	29
9.	SEGMENTOS	30
9.1	EXTENTS	30
9.2	BLOCKS	31
10.	SCHEMA	32
10.1.1	ESPAÇO LÓGICO	32
10.2	ISOLAMENTO	32
10.3	ORGANIZAÇÃO	33
10.4	OPERAÇÕES	33
10.5	COMPARTILHAMENTO	33
10.6	SCHEMA VS OWNER	33

1. DATABASE



Um banco de dados é um conjunto de arquivos físicos que armazenam **dados** e **metadados** do usuário. Os metadados consistem em informações estruturais, de configuração e de controle sobre o servidor de banco de dados. Você pode projetar seu banco de dados para ser um banco de

dados **Multitenant Database (CDB)** ou um banco de dados **non-cdb** (não CDB).

1.1 BANCO DE DADOS "NON-CDB"

Um banco de dados "non-CDB" (non-Container Database) refere-se ao modelo de arquitetura de banco de dados tradicional do Oracle, que existia antes da introdução do modelo de arquitetura "CDB" (Container Database) com a versão Oracle 12c.

Antes do Oracle 12c, todos os bancos de dados Oracle eram non-CDBs. Esse é o modelo clássico e tradicional de banco de dados que muitos DBAs e desenvolvedores estão acostumados.

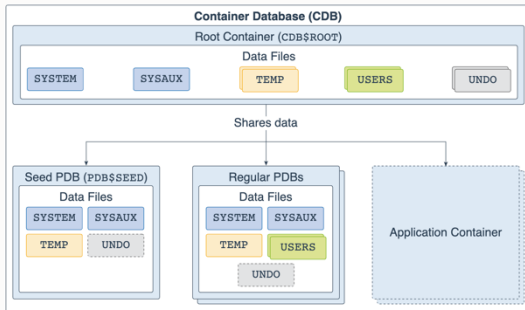
A grande maioria dos bancos ainda são non-cdb (não CDB), porém a partir da versão 21c não temos mais a opção de criar um database non-cdb, portanto, podemos dizer que a arquitetura non-cdb é o presente e a arquitetura cdb é o futuro (próximo)

Em um non-CDB, você tem uma única instância de banco de dados que contém todos os seus dados e objetos de esquema. Não há separação entre diferentes "containers" ou "pluggable databases".

A administração de um non-CDB é direta, pois não há necessidade de gerenciar múltiplos bancos de dados pluggable (PDBs) ou um container de banco de dados (CDB).



1.2 CONTAINER DATABASE (CDB)



Com a introdução do Oracle 12c, a Oracle introduziu o conceito de "Container Database" (CDB) e "Pluggable Database" (PDB). A ideia era permitir que múltiplos bancos de dados "pluggable" (PDBs) residissem dentro de um único banco de

dados container (CDB)

1.3 VANTAGENS DA ARQUITETURA MULTITENANT DATABASE

- **Consolidação:** Permite consolidar vários bancos de dados em um único banco de dados container, facilitando a administração e otimizando o uso de recursos.
- **Isolamento:** Cada PDB é isolado dos outros, o que significa que as operações em um PDB não afetam outros PDBs no mesmo CDB.
- **Mobilidade:** Os PDBs podem ser facilmente movidos, clonados ou realocados entre diferentes CDBs.
- **Administração Centralizada:** Embora haja vários PDBs, muitas tarefas administrativas podem ser realizadas no nível do CDB, afetando todos os PDBs contidos nele.

1.4 EXTRA OPTION

A arquitetura Multitenant Database é uma extra option. Ou seja, há custos adicionais para a sua utilização, caso você utilize **acima do limite** que é oferecido sem custos adicionais.

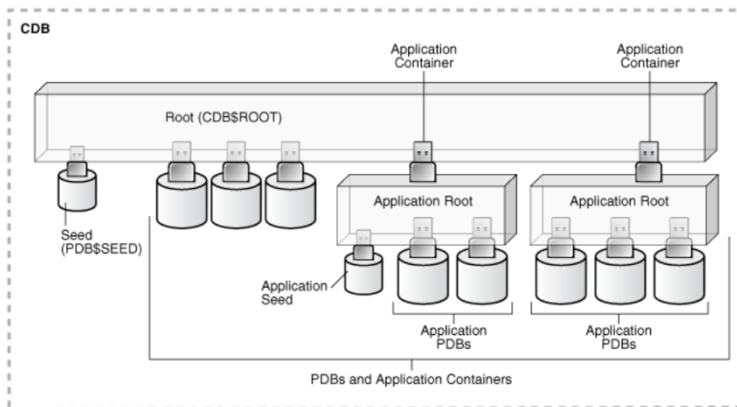
Se você não licenciar o Multitenant Database, é possível usar até **três Pluggable Database** por container.

https://docs.oracle.com/en/database/oracle/oracle-database/23/dbiad/db_cdb.html



2. MULTITENANT CONTAINER DATABASE

Figure 1-1 Containers in a CDB



O Multitenant Container Database (CDB) é um recurso introduzido no Oracle Database 12c que permite que um único banco de dados Oracle contenha múltiplos "Pluggable Databases" (PDBs).

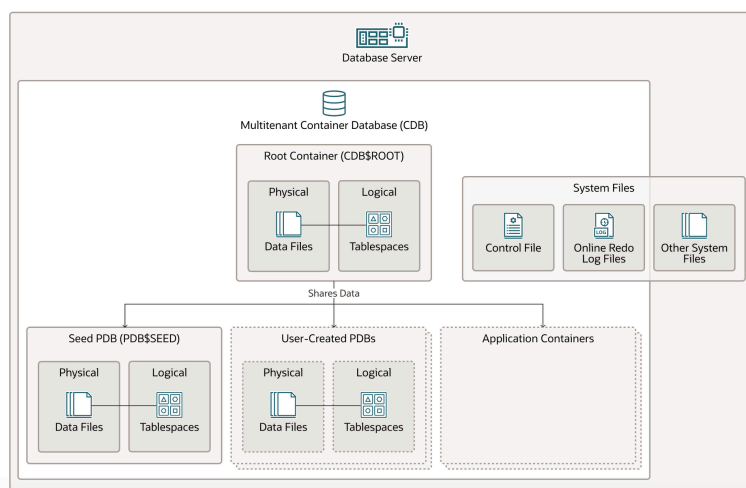
Container Database (CDB): É o banco de dados principal que hospeda os metadados comuns e as estruturas do sistema.

Ele contém os arquivos de dados, control files, redo log files, entre outros, que são compartilhados por todos os PDBs dentro dele.

Pluggable Database (PDB): É um banco de dados que reside dentro de um CDB. Cada PDB é uma coleção de esquemas, objetos e dados não compartilhados.

Eles aparecem e funcionam para os usuários finais como um banco de dados Oracle tradicional não-CDB, mas são na verdade apenas uma parte do CDB.

Multitenant Container Database (CDB)

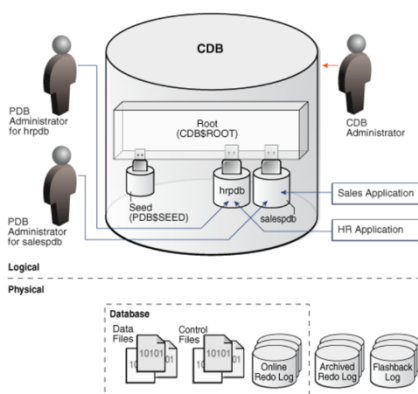


2.1 VANTAGENS

- **Consolidação:** O Multitenant permite consolidar vários bancos de dados em um único CDB, o que pode resultar em economia de recursos e facilidade de gerenciamento.
- **Isolamento:** Cada PDB é isolado dos outros, o que significa que um problema em um PDB não afeta os outros.
- **Rápida Implantação:** PDBs podem ser rapidamente provisionados e clonados. Se você tiver um PDB configurado exatamente como você gosta, pode cloná-lo rapidamente para criar um novo ambiente.
- **Gerenciamento Simplificado:** Operações como backup, recuperação e patching podem ser realizadas no nível do CDB, afetando todos os PDBs contidos nele.
- **Mobilidade:** Os PDBs podem ser facilmente movidos entre CDBs, permitindo uma grande flexibilidade na gestão e migração de bancos de dados.

2.2 USO

Figure 1-2 CDB with No Application Containers



O recurso Multitenant é particularmente útil para empresas que desejam consolidar seus bancos de dados, provedores de serviços em nuvem ou qualquer organização que deseje simplificar o gerenciamento de múltiplos bancos de dados.

Embora o Oracle Database 19c e versões posteriores permitam a criação de um CDB com um até 3 PDBs sem custo adicional, criar mais do que 3 PDBs,

requer a aquisição de uma Extra Option.

2.3 ESTRUTURA

Root Container (CDB\$ROOT)

- Contém metadados fornecidos pela Oracle e usuários comuns.
- Não armazena dados do usuário.
- Todos os PDBs pertencem ao Root Container.

Seed PDB (PDB\$SEED)

- É um template que o CDB usa para criar novos PDBs.
- Não pode ser modificado.

User-Created PDBs

- São PDBs criados pelo usuário.
- Cada PDB existe em isolamento completo dos outros PDBs no mesmo CDB.

2.4 REFERÊNCIAS

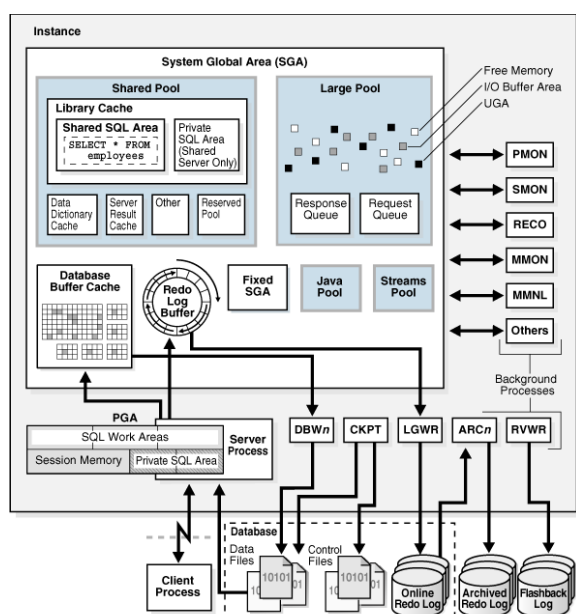
<https://docs.oracle.com/en/database/oracle/oracle-database/19/multi/introduction-to-the-multitenant-architecture.html#GUID-267F7D12-D33F-4AC9-AA45-E9CD671B6F22>

3. ESTRUTURAS DE ARMAZENAMENTO

Quando pensamos na estrutura de armazenamento do Oracle, podemos dividi-la de duas formas:

1. Estrutura de armazenamento física
2. Estrutura de armazenamento lógica

Isso permite gerenciar o armazenamento físico sem afetar o acesso às estruturas lógicas. Por exemplo, caso você mude o nome de um arquivo de dados ou mova ele para outro caminho, essa ação não afetará os dados armazenados neste arquivo.



A imagem ao lado mostra a relação entre as estruturas de memória da instância com os arquivos do banco de dados.

3.1 ESTRUTURA DE ARMAZENAMENTO FÍSICA

A estrutura de armazenamento física refere-se aos arquivos do banco de dados visíveis a nível de Sistema Operacional. Como por exemplo:

- Data Files
- Control Files
- Redo Log Files

3.2 ESTRUTURA DE ARMAZENAMENTO LÓGICA

Quando tratamos da estrutura de armazenamento lógica nos referimos a uma estrutura criada e organizada pelo próprio Oracle. Esta estrutura não é reconhecida

e visível a nível de Sistema Operacional. De forma resumida, essa estrutura é composta por:

- Tablespaces
- Segmentos
- Extents
- Blocks

<https://docs.oracle.com/en/database/oracle/oracle-database/23/cncpt/oracle-database-storage-structures.html#GUID-B679FE3E-8FC1-404C-A368-5A4AC0553664>

4. ESTRUTURA DE ARMAZENAMENTO FÍSICA

4.1 ARQUIVOS DE DADOS

Um arquivo de dados (data file) é uma das principais estruturas de armazenamento físico em um banco de dados Oracle.

Um arquivo de dados é um arquivo físico no sistema de armazenamento persistente. Ele contém estruturas de dados lógicas, como tabelas, índices, clusters e outros objetos de banco de dados.

Cada arquivo de dados está associado a um tablespace específico. Um tablespace pode consistir em um ou mais arquivos de dados.

O tablespace serve como uma camada lógica entre os objetos do banco de dados e os arquivos físicos de dados, permitindo uma gestão mais flexível do armazenamento.

Arquivos de dados podem ser configurados para crescer automaticamente à medida que o espaço é necessário.

Isso é útil para garantir que o banco de dados não fique sem espaço. Além disso, os DBAs podem redimensionar arquivos de dados manualmente ou adicionar mais arquivos de dados a um tablespace, se necessário.

Os DBAs podem realizar várias operações em arquivos de dados, como criar, mover, renomear, redimensionar, fazer backup e restaurar.

Essas operações são essenciais para a manutenção e gestão do espaço de armazenamento do banco de dados.

<https://docs.oracle.com/en/database/oracle/oracle-database/19/cncpt/physical-storage-structures.html#GUID-008A1F08-9C75-4E9F-A70B-41FB942C60B4>

4.2 ARQUIVOS TEMPORÁRIOS

Os tempfiles são arquivos especiais no Oracle Database que pertencem a tablespaces temporários e são usados para armazenar dados temporários, como os gerados durante operações de classificação e junção.

Eles desempenham um papel fundamental em muitas operações que requerem espaço de armazenamento temporário.

Os tempfiles são frequentemente usados em operações que necessitam de espaço temporário, como classificações, junções, criação de índices e operações de GROUP BY.

Se o espaço na memória (área de classificação na memória) não for suficiente para concluir a operação, o Oracle usará o tablespace temporário e, consequentemente, os tempfiles.

Os tempfiles podem ser criados quando um tablespace temporário é criado ou podem ser adicionados a um tablespace temporário existente.

Eles podem ser redimensionados, movidos ou excluídos como arquivos de dados regulares, mas as operações relacionadas a registros de redo (como backup e recuperação) não se aplicam a eles.

Os DBAs podem monitorar o uso de espaço em tempfiles para garantir que o espaço temporário seja adequado para as operações do banco de dados.

4.3 ARQUIVOS DE CONTROLE

Um control file é um componente vital do banco de dados Oracle que mantém o registro de todas as estruturas do banco de dados. Ele desempenha um papel crucial na arquitetura do banco de dados e é essencial para a operação e recuperação do banco de dados.

Um control file é um pequeno arquivo binário que registra a estrutura física do banco de dados. Ele contém informações sobre o banco de dados e sobre a estrutura e inicialização do Oracle.

Conteúdo do Control File

- Nome do banco de dados e identificador único do banco de dados (DBID).
- Informações sobre arquivos de dados, arquivos de redo log online e arquivos de redo log arquivados.
- Informações sobre tablespaces.
- Informações sobre backups realizados com o RMAN (Oracle Recovery Manager).
- SCN (System Change Number) de checkpoints, que indica até que ponto os dados foram salvos no disco.

Funções Principais

- **Registro de Estrutura:** O control file mantém um registro de todos os arquivos de dados e redo log files, ajudando o Oracle a localizar e gerenciar esses arquivos.
- **Recuperação:** O control file é crucial para a recuperação do banco de dados. Ele contém informações sobre o SCN, que é vital para determinar a partir de que ponto iniciar a recuperação.
- **Integridade:** O control file ajuda a garantir a integridade do banco de dados, mantendo um registro de todas as alterações estruturais.

4.3.1 MUITPLEXAÇÃO

O arquivo de controle é fundamental para o banco de dados, por isso, é importante que seja feita a Muitplexação do mesmo, que consiste em manter cópias idênticas do mesmo arquivo, de preferência em file systems diferentes:

- Para proteção contra falhas, o Oracle permite que múltiplas cópias idênticas de um control file sejam mantidas. Essas cópias são atualizadas

simultaneamente e podem ser armazenadas em diferentes dispositivos de armazenamento para fornecer redundância.

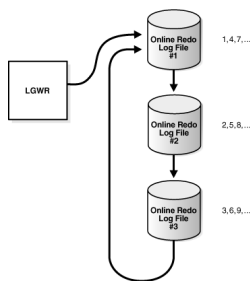
- Se um control file se tornar inacessível ou corrompido, o banco de dados pode continuar a operar usando uma das cópias redundantes.

O control file é acessado continuamente durante a operação do banco de dados e deve estar disponível para gravação sempre que o banco de dados estiver aberto. Embora o control file seja pequeno, sua importância é imensa, e a perda de todos os control files sem um backup adequado pode resultar em perda de dados.

Os DBAs devem monitorar e fazer backup regularmente dos control files.

<https://docs.oracle.com/en/database/oracle/oracle-database/19/cncpt/physical-storage-structures.html#GUID-90EAC9F2-B396-4C59-9821-B03BF7B6BCD0>

4.4 REDO LOG



Os redo log files são componentes cruciais do Oracle Database que registram todas as alterações feitas nos dados, permitindo a recuperação do banco de dados em caso de falhas. Eles são essenciais para garantir a durabilidade e integridade dos dados.

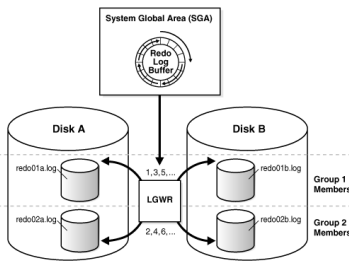
Os redo log files no Oracle Database desempenham um papel vital na garantia da durabilidade e recuperação dos dados. Eles registram todas as alterações feitas no banco de dados, permitindo que o banco de dados seja restaurado a um estado consistente após falhas.

A principal função dos redo log files é permitir a recuperação do banco de dados em caso de falhas, como falhas do sistema, falhas de instância ou corrupções.

Eles garantem que todas as transações confirmadas sejam preservadas e aplicadas, mesmo que os dados não tenham sido salvos nos arquivos de dados no momento da falha.



4.4.1 MULTIPLEXAÇÃO



A multiplexação envolve manter várias cópias idênticas dos redo logs para proteger contra a perda de dados devido a falhas de disco ou corrupção de arquivos.

Cada cópia é chamada de "membro" e um conjunto de membros idênticos forma um "grupo" de redo logs.

A ideia por trás da multiplexação é evitar um único ponto de falha. Se uma cópia (membro) de um redo log ficar corrompida ou inacessível devido a uma falha de disco, o Oracle ainda pode continuar operando usando uma das outras cópias.

4.4.2 REDO LOG SWITCH

Um log switch ocorre quando o Oracle Database para de escrever em um redo log e começa a escrever em outro. Isso geralmente acontece quando o redo log atual está cheio e o Oracle precisa continuar registrando as alterações.

Os redo logs são usados em uma circular. Quando o último redo log é preenchido, o Oracle retorna ao primeiro redo log e começa a escrever nele novamente.

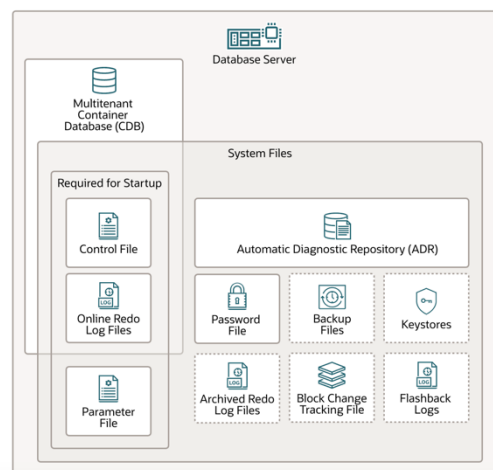
No entanto, antes que um redo log possa ser reutilizado, ele deve ser arquivado (se o banco de dados estiver no modo ARCHIVELOG).

<https://docs.oracle.com/en/database/oracle/oracle-database/19/cncpt/physical-storage-structures.html#GUID-47557C86-E551-46B5-B28E-28D6C500694E>



5. DATABASE SYSTEM FILES

Database System Files



O Oracle Database utiliza diversos arquivos de sistema que residem no servidor do banco de dados. Estes são diferentes dos arquivos de dados, que são arquivos físicos pertencentes aos contêineres do banco de dados.

5.1 ARQUIVOS DE CONTROLE

Um control file é um componente vital do banco de dados Oracle que mantém o registro de todas as estruturas do banco de dados. Ele desempenha um papel crucial na arquitetura do banco de dados e é essencial para a operação e recuperação do banco de dados. Um control file é um pequeno arquivo binário que registra a estrutura física do banco de dados. Ele contém informações sobre o banco de dados e sobre a estrutura e inicialização do Oracle.

5.2 PARAMETER FILE

O **Parameter File** é um arquivo essencial usado pelo Oracle Database para determinar como uma instância do banco de dados será inicializada e configurada ao ser iniciada. Ele contém uma lista de parâmetros de inicialização que definem características e comportamentos específicos da instância do banco de dados, como a quantidade de memória a ser alocada, o local dos arquivos de controle, o tamanho do buffer de cache, entre outros.



Existem dois tipos principais de arquivos de parâmetros no Oracle:

1. **Initialization Parameter File (pfile):** É um arquivo de texto simples que pode ser editado manualmente. Ele contém parâmetros de inicialização e seus valores.
2. **Server Parameter File (spfile):** É um arquivo binário que não pode ser editado diretamente. No entanto, ele pode ser modificado dinamicamente enquanto o banco de dados está em execução, usando comandos SQL.

5.3 ARCHIVE LOG

O Archive Log, é um componente fundamental do Oracle Database que armazena um registro histórico de todas as alterações feitas no banco de dados. Ele desempenha um papel crucial na garantia da durabilidade dos dados e na capacidade de recuperação do banco de dados.

Registro Histórico: O Archive Log contém um registro contínuo de todas as transações e alterações feitas no banco de dados. Cada entrada no Archive Log representa uma alteração específica feita nos dados.

Recuperação de Dados: Em caso de falha do sistema ou corrupção de dados, os Archive Logs são essenciais para restaurar o banco de dados ao seu estado anterior. Eles permitem que os administradores de banco de dados "rejoguem" transações passadas para recuperar dados perdidos ou corrompidos.

5.3.1 MODO ARCHIVELOG

Para que o Oracle Database gere Archive Logs, o banco de dados deve estar operando no modo ARCHIVELOG. Neste modo, após um redo log file ser preenchido com informações de transações, ele é arquivado como um Archive Log antes de ser reutilizado. Os Archive Logs são vitais para operações de backup e recuperação.

Eles permitem backups consistentes do banco de dados enquanto ele está online e são essenciais para a recuperação ponto a ponto, onde o banco de dados é restaurado a um estado específico no tempo.

Os DBAs devem gerenciar os Archive Logs, garantindo que haja espaço suficiente para armazená-los e decidindo por quanto tempo eles devem ser retidos. Também é comum mover Archive Logs mais antigos para um armazenamento de longo prazo ou excluí-los após terem sido usados para backups.

Em resumo, o Archive Log é um componente crucial do Oracle Database que permite a recuperação de dados e garante a durabilidade das transações.

5.4 AUTOMATIC DIAGNOSTIC REPOSITORY (ADR)

O Automatic Diagnostic Repository (ADR) é um repositório baseado em arquivos introduzido no Oracle Database para centralizar e simplificar o gerenciamento de dados de diagnóstico.

Ele desempenha um papel fundamental na identificação e resolução de problemas no banco de dados e em outros componentes do Oracle.

Estrutura Unificada: O ADR possui uma estrutura de diretório unificada, permitindo que múltiplas instâncias e produtos Oracle armazenem seus dados de diagnóstico em um local centralizado.

Isso facilita a localização e análise de informações relevantes quando surgem problemas.

Diversos Dados de Diagnóstico: O ADR armazena uma variedade de informações de diagnóstico, incluindo traces, dumps, o alert log, relatórios do health monitor, etc. Esses dados são essenciais para analisar e resolver problemas que podem surgir no banco de dados ou em componentes associados.

Integração com Outros Produtos Oracle: Além do banco de dados Oracle, outros produtos e componentes, como o Oracle Automatic Storage Management (Oracle

ASM), o listener e o Oracle Clusterware, também utilizam o ADR para armazenar seus dados de diagnóstico.

Facilita a Resolução de Problemas: Com o ADR, os administradores de banco de dados e os engenheiros de suporte da Oracle podem rapidamente acessar e analisar os dados de diagnóstico necessários para identificar a causa raiz de um problema e implementar uma solução.

Gerenciamento: O ADR é gerenciado usando ferramentas e utilitários específicos, como o ADRCI (Automatic Diagnostic Repository Command Interface). Com o ADRCI, os usuários podem visualizar alert logs, gerenciar incidentes, criar pacotes de incidentes para enviar à Oracle Support e realizar outras tarefas relacionadas ao diagnóstico.

Em resumo, o Automatic Diagnostic Repository (ADR) é uma ferramenta essencial no ecossistema Oracle, projetada para centralizar e simplificar o gerenciamento de dados de diagnóstico. Ele ajuda os administradores e engenheiros de suporte a identificar rapidamente problemas e implementar soluções, garantindo a operação contínua e eficiente do banco de dados e de componentes associados.

5.5 PASSWORD FILE

O Password File é um arquivo especial no Oracle Database que armazena senhas para usuários que têm privilégios administrativos, permitindo-lhes conectar-se ao banco de dados mesmo quando ele não está aberto.

Esse arquivo é crucial para a segurança e administração do banco de dados, pois garante que apenas usuários autorizados com os devidos privilégios possam realizar tarefas administrativas.

Os usuários com privilégios como SYSDBA, SYSOPER, SYSBACKUP, SYSDG, SYSKM, SYSRAC e SYSASM, por exemplo, precisam ser autenticados usando o Password File quando tentam se conectar ao banco de dados de forma remota.

O arquivo de senha ajuda a garantir que esses privilégios elevados não sejam abusados e que apenas indivíduos autorizados possam realizar operações sensíveis no banco de dados.

5.6 BLOCK CHANGE TRACK FILE

O Block Change Tracking File é um recurso no Oracle Database que facilita e acelera a realização de backups incrementais.

Esse arquivo mantém um registro dos blocos de dados que foram modificados desde o último backup, permitindo que o Oracle Recovery Manager (RMAN) identifique rapidamente quais blocos precisam ser incluídos em um backup incremental.

Em vez de examinar todos os blocos de dados para determinar quais foram alterados, o RMAN pode consultar o Block Change Tracking File para identificar apenas os blocos que foram modificados.

Isso resulta em backups incrementais muito mais rápidos, pois o RMAN precisa ler e fazer backup de um número significativamente menor de blocos.

O uso do Block Change Tracking File melhora a eficiência dos backups incrementais, reduzindo o tempo e os recursos necessários para completar o backup.

No entanto, é importante observar que o arquivo de rastreamento de alterações de blocos é opcional, mas quando habilitado, oferece benefícios significativos em termos de desempenho de backup.

5.7 FLASHBACK DATABASE

O Flashback Database é um recurso do Oracle Database que permite reverter o banco de dados inteiro para um estado anterior no tempo, sem a necessidade de restaurações tradicionais de backup.

Esse recurso é extremamente útil para recuperação rápida de erros, como exclusões acidentais ou alterações indesejadas.

Os arquivos de Flashback Database, conhecidos como flashback logs, desempenham um papel crucial nesse processo.

Eles armazenam informações sobre as alterações nos blocos de dados do banco de dados. Esses logs são usados para capturar e salvar imagens de blocos de dados alterados em intervalos regulares.

5.8 REFERÊNCIAS

https://docs.oracle.com/en/database/oracle/oracle-database/23/dbiad/db_dbfiles.html

6. LOCAIS DE ARMAZENAMENTO DOS DATABASE FILES

Os arquivos do banco de dados podem ser armazenados de formas diferentes, as mais comuns são:

- File System
- ASM

6.1.1 FILE SYSTEM

```
SQL> select name from v$datafile;

NAME
-----
/u02/oradata/FREE/FREEPDB1/system01.dbf
/u02/oradata/FREE/FREEPDB1/sysaux01.dbf
/u02/oradata/FREE/FREEPDB1/undotbs01.dbf
/u02/oradata/FREE/FREEPDB1/users01.dbf
```

A maioria dos bancos Oracle, armazenam os arquivos em um file system tradicional de Sistema Operacional, principalmente quando se trata de ambientes Oracle Standard.

É uma forma simples de se gerenciar que permite uma visibilidade clara da localização dos arquivos, assim como o acesso a eles.

```
[oracle@vm01ol8193 ~]$ ls -ltr /u02/oradata/FREE/FREEPDB1/
total 989M
-rw-r-----. 1 oracle oinstall 5.1M Apr 23 18:03 users01.dbf
-rw-r-----. 1 oracle oinstall 21M Apr 23 22:00 temp01.dbf
-rw-r-----. 1 oracle oinstall 301M Apr 24 01:03 system01.dbf
-rw-r-----. 1 oracle oinstall 101M Apr 24 01:03 undotbs01.dbf
-rw-r-----. 1 oracle oinstall 581M Apr 24 01:03 sysaux01.dbf
```

6.1.2 ASM

O ASM (Automatic Storage Management) é um recurso do Oracle Database que oferece um sistema de gerenciamento de arquivos e volumes simplificado, especialmente projetado para bancos de dados Oracle. Ele elimina a necessidade de depender de sistemas de arquivos do sistema operacional convencionais ou de gerenciadores de volumes de terceiros para armazenar arquivos de dados, arquivos de log e outros arquivos relacionados ao banco de dados. O ASM proporciona uma maneira eficiente e integrada de gerenciar todos os arquivos do banco de dados em diferentes tipos de armazenamento.

Para utilizar o ASM, é preciso instalar o **Oracle Clustware**, normalmente é utilizado em ambientes **Oracle RAC** ou em ambiente com um **grande volume de dados**.

6.2 ORACLE MANAGED FILES

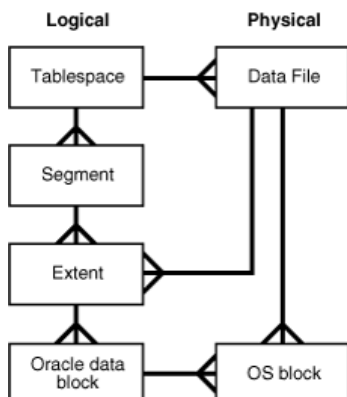
O Oracle Managed Files (OMF) é um recurso aonde o próprio Oracle Database gerencia o nome e localização dos arquivos do banco de dados. Por exemplo, ao criar um tablespace, não é necessário especificar os nomes dos arquivos de dados.

O Oracle gera automaticamente um nome de arquivo único e adequado. Isso simplifica muito a administração do banco de dados. Este recurso não é obrigatório quando você armazena os arquivos dentro de um file system, mas se você armazenar os arquivos no ASM, o OMF é obrigatório.

```
NAME
-----
/u02/oradata/FREE/FREEPDB1/system01.dbf
/u02/oradata/FREE/FREEPDB1/sysaux01.dbf
/u02/oradata/FREE/FREEPDB1/undotbs01.dbf
/u02/oradata/FREE/FREEPDB1/users01.dbf
/u02/oradata/FREE/166D2C9788C03584E0650A0027C36979/datafile/o1_mf_x_m2k3bsqj_.dbf
```

Quando você opta por especificar o caminho completo do arquivo, chamamos esse processo de **User-Managed Files**, pois você gerencia diretamente os arquivos do sistema operacional utilizados pelo banco. Isso inclui tomar decisões sobre a estrutura e nomeação dos arquivos. Por exemplo, ao criar um tablespace, o administrador especifica o nome e o caminho dos arquivos de dados.

7. ESTRUTURA LÓGICA DO BANCO DE DADOS



A estrutura de armazenamento lógico refere-se aos componentes lógicos do Oracle Database que determinam como os dados são armazenados, organizados e gerenciados, independentemente de sua representação física real nos dispositivos de armazenamento. Um database armazena objetos em arquivos de dados.

Essas estruturas lógicas permitem que o banco de dados gerencie os dados de maneira flexível e eficiente, sem se preocupar diretamente com detalhes físicos específicos.

As principais componentes da estrutura de armazenamento lógico incluem:

- Tablespaces
- Segmentos
- Extents
- Blocks (Blocos)

A estrutura de armazenamento lógico do Oracle Database define como os dados são organizados e gerenciados em níveis lógicos, permitindo que o banco de dados opere de maneira eficiente e flexível.

Essas estruturas lógicas **abstraem os detalhes físicos reais do armazenamento**, permitindo que os administradores e desenvolvedores trabalhem com dados sem se preocupar com a localização física exata ou a organização dos dados no disco.

7.1 REFERÊNCIAS

<https://docs.oracle.com/en/database/oracle/oracle-database/19/cncpt/oracle-database-storage-structures.html#GUID-B679FE3E-8FC1-404C-A368-5A4AC0553664>



8. TABLESPACES

Uma tablespace é uma unidade lógica de armazenamento no Oracle Database que agrupa um ou mais arquivos de dados físicos e serve como contêiner para armazenar estruturas de dados, como tabelas e índices. Ela permite a organização, separação e gerenciamento dos dados de maneira flexível dentro do banco de dados.

- Contêm zero ou mais segmentos correspondentes a objetos de esquema.
- Podem conter segmentos alocados para diferentes esquemas.
- Cada PDB e contêiner de aplicativo tem seu próprio conjunto de tablespaces.
- Você pode armazenar dados do usuário no tablespace USERS padrão ou criar tablespaces adicionais.

8.1 SYSTEM TABLESPACE

Uma SYSTEM tablespace é uma tablespace essencial no Oracle Database que contém metadados, incluindo dicionário de dados do sistema e objetos de sistema.

Ela é criada automaticamente durante a instalação do banco de dados e é fundamental para o funcionamento do Oracle Database.

Contém tabelas e views com informações administrativas sobre o banco de dados e objetos compilados, e pacotes.

8.2 SYSAUX TABLESPACE

A tablespace SYSAUX é uma tablespace auxiliar à tablespace SYSTEM, introduzida no Oracle 10g. Ela armazena informações de diversos componentes e recursos do Oracle Database, desempenhando um papel fundamental para o funcionamento do banco de dados.

Ela ajuda a descongestionar a tablespace SYSTEM, armazenando componentes e recursos que, em versões anteriores do Oracle, poderiam estar na SYSTEM. Isso torna o gerenciamento e a manutenção do banco de dados mais eficientes.

Alguns componentes da tablespace SYSAUX:

- **Estatísticas de Otimização:** Dados coletados pelo Oracle Optimizer para otimizar o desempenho das consultas
- **Informações de AWR (Automatic Workload Repository):** Estatísticas e métricas de desempenho coletadas automaticamente.
- **Informações de Jobs:** Dados relacionados a jobs agendados no banco de dados.

8.3 UNDO TABLESPACE

A tablespace UNDO é uma tablespace especial no Oracle Database que armazena informações de "rollback" (undo). Ela permite que o banco de dados reverta transações, forneça leitura consistente e suporte a recuperação de instâncias. Dentro da tablespace UNDO, encontramos:

Segmentos de Desfazer (Undo Segments): Estes segmentos contêm os registros de rollback, que são históricos das alterações feitas nos dados.

Temos dois modos de configuração para tablespace de UNDO em ambientes CDB:

1. **Local Undo Mode:** Um undo tablespace em cada PDB.
2. **Shared Undo Mode:** Um undo tablespace ativo em um CDB de instância única ou um para cada instância em um Oracle RAC CDB.

8.4 USER TABLESPACE

São tablespaces criadas explicitamente pelos administradores de banco de dados (DBAs) ou outros usuários com os privilégios necessários para atender a requisitos específicos de armazenamento e organização de dados.

Diferentemente das tablespaces do sistema, como SYSTEM e SYSAUX, que são criadas automaticamente durante a instalação do Oracle Database, as tablespaces

criadas pelo usuário são definidas para armazenar dados de aplicativos, objetos de esquema e outros dados específicos do usuário.

8.5 TEMPORARY TABLESPACE

É uma tablespace especializada no Oracle Database utilizada para armazenar dados temporários, como os gerados durante operações de classificação e junção.

Ao contrário das tablespaces permanentes, que armazenam dados persistentes, as tablespaces temporárias não contêm objetos de esquema permanentes e os dados nelas são descartados após o término da sessão ou transação que os utilizou.

Elas são essenciais para operações que requerem espaço de armazenamento temporário, garantindo que o desempenho do banco de dados não seja comprometido ao realizar tarefas que geram grandes volumes de dados intermediários.

8.6 BIGFILE TABLESPACE

Bigfile tablespaces são um tipo de tablespace no Oracle Database que contém um único, mas potencialmente muito grande, arquivo de dados. Ao contrário das tablespaces tradicionais, que podem ter múltiplos arquivos de dados, cada bigfile tablespace é simplificada para ter apenas um.

Bigfile tablespaces são ideais para bancos de dados de grande porte ou ambientes que buscam simplificar o gerenciamento de armazenamento e maximizar a capacidade de armazenamento. Abaixo algumas vantagens:

- **Simplificação de Gerenciamento:** Com apenas um arquivo de dados por tablespace, o gerenciamento e a administração tornam-se mais simples.
- **Capacidade de Armazenamento Ampliada:** Bigfile tablespaces podem suportar arquivos de dados muito grandes, aproveitando sistemas de arquivos de grande capacidade e volumes de armazenamento.
 - **Menor Overhead de Gerenciamento:** Com menos arquivos de dados para gerenciar, há uma redução no overhead associado ao gerenciamento de múltiplos arquivos.

8.7 SMALLFILE TABLESPACE

Smallfile tablespaces são o tipo tradicional de tablespace no Oracle Database e podem conter múltiplos arquivos de dados, mas cada arquivo tem um tamanho limitado.

Smallfile tablespaces oferecem uma abordagem mais tradicional e granular para o gerenciamento de armazenamento no Oracle Database, proporcionando flexibilidade e controle detalhado para os administradores de banco de dados. Abaixo algumas vantagens:

- **Flexibilidade:** Smallfile tablespaces permitem uma distribuição mais granular dos dados, já que os administradores podem espalhar os arquivos de dados em diferentes dispositivos ou sistemas de arquivos, se necessário.
- **Gerenciamento Familiar:** Para DBAs que trabalharam com versões anteriores do Oracle Database, o gerenciamento de smallfile tablespaces é familiar e bem compreendido.
- **Maior Controle:** Com múltiplos arquivos de dados, os DBAs têm mais controle sobre a localização física dos dados e podem otimizar o desempenho, colocando arquivos de dados específicos em dispositivos de armazenamento de alto desempenho.
- **Recuperação Granular:** Em caso de falha de um arquivo de dados, apenas esse arquivo específico precisa ser recuperado, o que pode ser mais rápido do que recuperar um arquivo muito grande em bigfile tablespaces.

8.8 ARQUIVOS FÍSICOS

Tablespaces no Oracle Database são unidades lógicas de armazenamento, mas, para armazenar dados efetivamente, elas dependem de arquivos físicos no sistema de armazenamento subjacente.

Estes arquivos físicos são os componentes reais no disco que contêm os dados e metadados associados a uma tablespace

As tablespaces no Oracle Database fornecem uma abstração lógica para o armazenamento de dados, mas, por trás dessa abstração, estão os arquivos físicos que efetivamente armazenam e gerenciam os dados no disco.

Estes arquivos são vitais para a operação, manutenção e recuperação do banco de dados.

8.9 REFERÊNCIAS

<https://docs.oracle.com/en/database/oracle/oracle-database/19/admin/managing-tablespaces.html#GUID-1C162C60-6698-44F2-B2A9-F3E2D2958D88>

9. SEGMENTOS

Um segmento é uma estrutura no Oracle Database que representa um conjunto de dados, como uma tabela ou índice. Ele é formado por blocos, que são unidades básicas de armazenamento, e fica dentro de uma tablespace.

Quando você cria uma tabela ou índice, o Oracle reserva um segmento para ela. Esse segmento é onde os dados da tabela serão armazenados.

<https://docs.oracle.com/en/database/oracle/oracle-database/23/cncpt/logical-storage-structures.html#GUID-7DA83E64-9FF1-45A7-A9AC-D4997DDE0866>

9.1 EXTENTS

Um "extent" é um conjunto de blocos contíguos (ou seja, blocos que estão lado a lado) em um banco de dados Oracle. Esses blocos são alocados para armazenar dados específicos.

Quando um segmento é criado ou precisa crescer porque os dados estão sendo adicionados, o Oracle aloca um novo "extent" para esse segmento. Portanto, um segmento é composto por um ou mais "extents". À medida que o segmento cresce, mais "extents" são adicionados a ele.

À medida que os dados são inseridos na tabela, eles são armazenados em extents dentro do segmento da tabela. Se um extent ficar cheio, o Oracle automaticamente aloca outro extent dentro do segmento para armazenar mais dados.

<https://docs.oracle.com/en/database/oracle/oracle-database/23/cncpt/logical-storage-structures.html#GUID-E3D8E030-A056-40AC-9B7E-8C957E28EE75>

9.2 BLOCKS

Um "bloco" é a **menor unidade de armazenamento** em um banco de dados Oracle. É onde os dados são efetivamente armazenados, como registros de uma tabela ou entradas de um índice.

Um "extent" é um conjunto de blocos contíguos. Em outras palavras, quando o Oracle precisa de mais espaço para armazenar dados e aloca um novo "extent", ele está, na verdade, reservando um grupo de blocos que estão lado a lado no armazenamento.

Portanto, enquanto o "bloco" é a unidade individual de espaço, o "extent" é como um "pacote" de blocos. E, como mencionado anteriormente, os segmentos são formados por um ou mais "extents".

<https://docs.oracle.com/en/database/oracle/oracle-database/23/cncpt/logical-storage-structures.html#GUID-1AED5140-E820-436C-BEB7-2A985524911E>

10. SCHEMA

Um "schema" no Oracle Database é uma **coleção lógica de objetos** de banco de dados. Estes objetos podem incluir tabelas, índices, views, sequências, sinônimos, procedimentos, funções, triggers, pacotes, tipos, entre outros.

Basicamente, qualquer estrutura que você possa criar e manipular com comandos SQL e PL/SQL pode residir dentro de um schema.

No Oracle, um schema está **diretamente associado** a um usuário do banco de dados. De fato, o nome do schema é o mesmo que o nome do usuário que o possui. Por exemplo, se você criar um usuário chamado "JOAO", um schema com o nome "JOAO" também será criado.

10.1 ESPAÇO LÓGICO

Um schema é uma entidade lógica, o que significa que não se refere a um espaço físico no disco ou a uma localização específica.

Em vez disso, refere-se a uma organização lógica de objetos. Os dados reais desses objetos são armazenados em tablespaces, que têm uma representação física no sistema de armazenamento.

Um schema no Oracle Database é uma coleção lógica de objetos de banco de dados que pertencem a um usuário específico.

Ele oferece uma maneira de organizar, isolar e gerenciar conjuntos de objetos relacionados, proporcionando flexibilidade, segurança e eficiência no gerenciamento e acesso aos dados.

10.2 ISOLAMENTO

Cada schema funciona como um espaço de trabalho separado e isolado. Os objetos em um schema são separados dos objetos em outro schema. Isso permite que diferentes usuários e aplicações trabalhem de forma independente, sem interferir uns nos outros.

10.3 ORGANIZAÇÃO

Os schemas permitem que os DBAs e desenvolvedores organizem e agrupem objetos relacionados juntos.

Por exemplo, todos os objetos relacionados a um aplicativo de folha de pagamento podem ser colocados no schema "FOLHA_PAGAMENTO", enquanto os objetos relacionados a um sistema de inventário podem residir no schema "INVENTARIO".

10.4 OPERAÇÕES

Os schemas podem ser alvo de várias operações, como exportação (usando ferramentas como o Data Pump Export) ou importação (usando o Data Pump Import).

Isso facilita a migração, backup ou duplicação de conjuntos lógicos de objetos de banco de dados.

10.5 COMPARTILHAMENTO

Embora cada objeto pertença a um schema específico, é possível criar sinônimos (aliases) para objetos em outros schemas.

Isso permite que um usuário acesse um objeto em um schema diferente como se estivesse em seu próprio schema.

10.6 SCHEMA VS OWNER

O owner é o usuário do banco de dados que possui (ou seja, tem os direitos sobre) um schema específico.

Em outras palavras, é o usuário que tem permissão para criar, modificar ou excluir objetos dentro de um schema específico.

A principal diferença entre os dois termos é a perspectiva de foco.

Quando falamos de "schema", estamos nos referindo à **coleção lógica de objetos** de banco de dados.

Quando falamos de "owner", estamos nos referindo ao **usuário que possui e controla esse schema**.

<https://docs.oracle.com/en/database/oracle/oracle-database/19/cncpt/introduction-to-oracle-database.html#GUID-35C20601-E266-486E-987B-7F355DB10DD4>