

# SUBP

---

Vježbe - Napredna administracija Oracle21c

# Sadržaj

- Konfiguracija RMAN
- Upotreba RMAN
- Database Recovery RMAN
- Flashback RMAN
- Monitoring and Managing Memory
- Database Schema Objects and Managing Storage
- Managing Resources
- Automatic Tasks With the Scheduler

# Konfiguracija RMAN

---

# Database Backups

- Da bi se odradio backup baze podataka koristi se alat pod nazivom Recovery Manager (RMAN)
- Backup strategija može biti
  - whole – cijela baza podataka se bekapuje (svi data fajlovi i bar jedan control fajl)
  - partial – dio baze podataka
- Backup tip može biti
  - full – sve informacije iz svih data fajlova
  - incremental – bekapuju se samo informacije koje su izmjenjene od prethodnog backup- a

# Database Backups

- Backup režim (mod) može biti
  - offline – konzistentan, backup se radi kada baza podataka nije u open režimu – data fajlovi su sinhronizovani sa control fajlovima
  - online – nekonzistentan, backup se radi kada je baza podataka u open režimu – data fajlovi nisu sinhronizovani sa control fajlovima i potreban je recovery nakon backup- a
- Backup može biti skladišten kao
  - image copies – duplikati data i archived log fajlova, prednost je u slučaju oporavka pojedinačnih fajlova
  - backup sets – predstavlja kopije jedan ili više data i archived log fajlova, prazni data blokovi se ne čuvaju. Mogu biti dodatno kompresovani, prednost je što zauzimaju manje prostora

# Konfiguracija RMAN

- RMAN (Recovery MANager) omogućava kreiranje i upravljanje backup-ima, detekciju loših (corrupted) blokova, kompresovanja podataka prilikom backup- a itd
- Potreban je RMAN Recovery Catalog koji predstavlja šemu baze podataka koje se koristi od strane RMAN- a da skladišti metapodatke o jednoj ili više Oracle baza podataka
- Tipično katalog treba da se skladišti u zasebnu bazu podataka

# Konfiguracija RMAN

- Recovery catalog obezbeđuje sledeće benefite:
  - recovery catalog obezbeđuje redundantnost za RMAN repozitorijum koji se skladišti u control file svake odredišne baze podataka. Recovery catalog služi kao sekundarni meta repository. Ako su odredišni control file i bekapi izgubljeni onda RMAN metapodaci još uvijek postoje u recovery katalogu.
  - recovery catalog centralizuje metapodatke za sve odredišne baze podataka. Skladištenje metapodataka u jedno mjesto olakšava izvještavanje i administracijske zadatke.
  - Recovery catalog može da skladišti istorijske metapodatke mnogo duže nego control file. Ova mogućnost je korisna ako treba da se uradi recovery koji ide u prošlost duže od istorijskih podataka u control file- u.

# Kreiranje Recovery Catalog- a

- Da bi se kreirao recovery catalog potrebno je:
  - konfigurirati bazu podataka koja sadrži recovery catalog
  - kreirati korisnika baze podataka koji je vlasnik recovery catalog- a
  - kreiranje samog recovery catalog- a

# Kreiranje Recovery Catalog- a

- Kreiranje baze podataka - logovati se kao sys u CDB putem SQL Developer- a
  - `create pluggable database catdb`  
`admin user catdbadmin identified by oracle;`
- Kreiranje korisnika
  - logovati se kao sys u CDB putem sqlplus- a
  - `alter session set container=catdb;`
  - `select status from v$instance;`
  - `startup open;`
  - `alter pluggable database catdb save state;`

# Kreiranje Recovery Catalog- a

- Kreiranje korisnika
  - logovati se kao sys u CATDB putem SQL Developer- a
  - create tablespace tools  
datafile  
'C:\oracle21c\_setup\app\oracle\oradata\ORCL\DATAFILE\tools01.dbf'  
size 1G autoextend on next 100M maxsize unlimited  
logging;

# Kreiranje Recovery Catalog- a

- Kreiranje korisnika
  - kreiranje šeme/korisnika za recovery katalog
  - `create user rco identified by oracle temporary tablespace temp default tablespace tools quota unlimited on tools;`
  - dodjeljivanje recovery catalog role korisniku
  - `grant recovery_catalog_owner to rco;`

# Kreiranje Recovery Catalog- a

- Kreiranje recovery catalog- a
  - konektovati se kroz rman u catalog bazu podataka (otvoriti cmd)
  - `rman catalog rco/oracle@localhost:1521/catdb`
  - `create catalog;`
  - `exit;`
  - logovati se kao rco u CATDB putem SQL Developer- a
  - `select table_name from user_tables;`

# Konektovanje na odredišnu bazu podataka

- Konektovanje na odredišnu bazu podataka u CMD izvršiti
  - `rman target sys/oracle@localhost:1521/orcl catalog rco/oracle@localhost:1521/catdb`
- Registrovanje baze podataka
  - `register database;`
- Prikaz šeme registrovane baze podataka
  - `report schema;`
  
- Ako se prilikom konekcije na target (odredišnu) bazu podataka izostavi catalog, rman upisuje podatke u control fajlove, a ne u katalog

# Konfigurisanje RMAN- a

- Prikaz svih konfiguracionih parametara (pregledati dobro sve parametre)
  - `show all;`
- Koristimo CONFIGURE naredbu da podesimo razne stavke za RMAN
- Podešavanje RMAN da prilikom izvršenja backup- a baze podataka automatski backup- uje control file i SPFILE
  - `configure controlfile autobackup on;`
  - `configure controlfile autobackup off;`
- Prikaz politike zadržavanja backup- a
  - `show retention policy;`
- Postoje dva tipa zadržavanja koji su isključivi
  - broj redundantnih kopija
  - zadržavanje backup- a N dana, prozor oporavka (recovery window)

## Konfigurisanje RMAN- a

- Podešavanje da postoje tri redundantne kopije backup- a
  - `configure retention policy to redundancy 3;`
- Podešavanje prozora oporavka na 15 dana
  - `configure retention policy to recovery window of 15 days;`
- Moguće je i da ne postoji politika zadržavanja
  - `configure retention policy to none;`

## Konfigurisanje RMAN- a

- Konfigurisanje RMAN da prilikom kreiranja backup- a ne uključi određeni tablespace
  - `configure exclude for tablespace tbs_name;`
  - `configure exclude for tablespace tbs_name clear;`
  - `show exclude;`
- Ako je u pitanju tablespace iz pdb baze podataka
  - `configure exclude for tablespace [pdb_name:]tbs_name;`
  - `configure exclude for tablespace [pdb_name:]tbs_name clear;`
  - `show exclude;`

## Konfigurisanje RMAN- a

- Isključiti inventorypdb tablespace iz backup- a (orclpdb1 pluggable baza podataka)
  - `configure exclude for tablespace orclpdb1:inventory;`
  - `show exclude;`
  - `configure exclude for tablespace orclpdb1:inventory clear;`
  - `show exclude;`

# Upotreba RMAN

---

# Backup

- Voditi računa prije početka da je zadat parametar `db_recovery_file_dest` (fast recovery area) jer se to koristi kao podrazumjevana destinacija za čuvanje backup- a
  - pogledati ranije lekcije
- Izvršiti backup data file- ova koji pripadaju tablespace- ovima `inventory` i `users` (u pluggable bazi `orclpdb1`)
  - `backup as backupset tablespace orclpdb1:users, orclpdb1:inventory;`
- Izvršiti backup pluggable baze podataka `orclpdb1`, backup treba da bude u formatu kompresovanog backup seta
  - `backup as compressed backupset pluggable database orclpdb1;`
- Izvršiti backup pluggable baze podataka `orclpdb1`, backup treba da bude image copy
  - `backup as copy pluggable database orclpdb1;`

# Backup

- Tagovi se koriste kao imena, odnosno reference za backup- e i mogu se kasnije koristiti uz razne naredbe
- Kreirati image kopiju tablespace- a system iz pluggable baze podataka orclpdb1 sa tagom system01
  - `backup as copy tablespace orclpdb1:system tag=system01;`
- Kreirati direktorijum C:\backup
- Kreirati image kopiju tablespace- a sysaux iz pluggable baze podataka orclpdb1 sa tagom sysaux01, naziv kopije treba da bude sysaux01.cpy
  - `backup as copy format 'C:\backup\sysaux01.cpy' tablespace orclpdb1:sysaux tag=sysaux01;`

# Backup

- Prilikom backup- a postoje dodatne opcije da se sačuvaju i archive log fajlovi kao i da se uključe svi tablespace- ovi čak i oni koji su exclude
- Izvršiti kompresovani backup pluggable baze podataka orclpdb1 koji uključuje i ARCHIVELOG fajlove i sve tablespace- ove koji pripadaju toj bazi podataka
  - `backup noexclude as compressed backupset pluggable database orclpdb1 plus archivelog;`
- Izvršiti incremental backup pluggable baze podataka orclpdb1 level 0
  - `backup incremental level 0 pluggable database orclpdb1;`

# Backup

- Izvršiti differential incremental backup pluggable baze podataka orclpdb1 level 1 - podrazumjevani tip incremental backup- a koji čuva sve izmjenjene blokove od poslednjeg level 1 ili level 0 incremental backup- a
  - `backup incremental level 1 pluggable database orclpdb1;`
- Efekti?
- Pregledati koliko prostora zauzima level 0, a koliko level 1!
- Izvršiti cumulative incremental backup pluggable baze podataka orclpdb1 level 1 - čuva sve blokove izmjenjene od poslednjeg level 0 incremental backup- a
  - `backup incremental level 1 cumulative pluggable database orclpdb1;`
- Efekti?

## List, report i delete naredba

- Prikazati kreirane backup- e pluggable baze podataka orclpdb1
  - `list backup of pluggable database orclpdb1;`
- Prikazati kreirane backup- e za tablespace inventory iz pluggable baze podataka orclpdb1
  - `list copy of tablespace orclpdb1:inventory;`
- Prikazati strukturu baze podataka kroz RMAN
  - `report schema;`
- Prikazati datafile- ove koji nisu backup- ovani
  - `report need backup;`
- Prikazati backup- e koji mogu biti obrisani
  - `report obsolete;`
- Obrisati backup- e koji su zastarjeli po retention policy
  - `delete obsolete;`

# Database recovery RMAN

---

## Restore i recovery

- Zapisati trenutno vrijeme (datum plus vrijeme)
- Logovati se kao sys u PDB putem SQL Developer- a i dodati 3 sloga u tabelu korisnik.radnik
  - `select count(*) from korisnik.radnik;`
  - `insert into korisnik.radnik values(korisnik.seq_mbr.nextval, 'A', 'B', 50000, 'Z');`
  - `commit;`
  - `select count(*) from korisnik.radnik;`
  - zapamtiti broj slogova prije i poslije dodavanja!

# Restore i recovery

- Zatvoriti sve konekcije nad pluggable bazom orclpdb1
- Logovati se kao sys u CDB putem sqlplus- a
  - prebaciti pluggable bazu podataka orclpdb1 u mount režim
  - `alter pluggable database orclpdb1 close;`
- Pokrenuti rman
  - `rman target sys/oracle@localhost:1521/orcl catalog rco/oracle@localhost:1521/catdb`

# Restore i recovery

- Napisati i izvršiti skriptu za restore i recovery (vrijeme prije dodavanja slogova)
  - run {

```
set until time "to_date('2023-12-13 09:01:00', 'YYYY-MM-DD  
HH24:MI:SS')";  
restore pluggable database orclpdb1;  
recover pluggable database orclpdb1;  
alter pluggable database orclpdb1 open resetlogs;  
}
```
- Provjeriti broj slogova u tabeli korisnik.radnik. Efekti?
- Moguće je uraditi restore i recovery putem SCN- a

# Flashback RMAN

---

# Flashback RMAN

- Provjera da li je flashback uključen
- Logovati se kao sys u CDB putem sqlplus- a
  - `select flashback_on from v$database;`
- Uključivanje flashback- a
- Logovati se kao sys u CDB putem sqlplus- a
  - `shutdown immediate;`
  - `startup mount;`
  - `alter database flashback on;`
  - `alter database open;`

# Flashback RMAN

- Prikazivanje tekućeg SCN- a (System Change Number)
  - `select current_scn from v$database;`
- Prikazivanje najmanjeg SCN- a na koji se može izvršiti flashback baze podataka
  - `select oldest_flashback_time, oldest_flashback_scn from v$flashback_database_log;`
- Prikazivanje procjenjene količine memorije potrebne za flashback da ispuni retention policy i količinu flashback podataka koji se čuvaju (u bajtovima)
  - `select estimated_flashback_size, flashback_size from v$flashback_database_log;`

# Flashback RMAN

- Logovati se kao sys u PDB putem SQL Developer- a i dodati 3 sloga u tabelu korisnik.radnik
  - `select count(*) from korisnik.radnik;`
  - `select current_scn FROM v$database; -> output: 32588508`
  - `insert into korisnik.radnik values(korisnik.seq_mbr.nextval, 'A', 'B', 50000, 'Z');`
  - `commit;`
  - `select count(*) from korisnik.radnik;`
  - `select current_scn FROM v$database; -> output: 32588519`

# Flashback RMAN

- Logovati se kao sys u CDB putem sqlplus- a
  - prebaciti pluggable bazu podataka orclpdb1 u mount režim
  - `alter pluggable database orclpdb1 close;`
  - `flashback pluggable database orclpdb1 to scn 32588508;`
  - `alter pluggable database orclpdb1 open resetlogs;`
- Provjeriti broj slogova u tabeli korisnik.radnik. Efekti?
- Ukoliko želimo uraditi flashback samo na konkretnoj PDB local undo režim mora biti uključen, ako taj režim nije uključen mora se raditi flashback na nivou CDB

# Monitoring and Managing Memory

---

# Monitoring and Managing Memory

- Upravljanje memorijom je veoma važno jer imamo ograničenu količinu memorije u sistemu
- Baferi imaju jedno od četiri stanja
  - pinned – trenutno se koristi čita/piše
  - clean – ne koristi se trenutno, očišćen nakon korišćenja
  - free – prazan je, nikad nije ni korišćen (npr. nakon pokretanja instance)
  - dirty – ne koristi se trenutno, ali još uvijek nije očišćen
- Od bafare u okviru Oracle SUBP- a se nalaze: shared pool, buffer cache, large pool i java pool

# Monitoring and Managing Memory

- Veličina bafera može da se vidi na sledeći način
- Logovati se kao sys u CDB putem SQL Developer- a
  - `select name, value from v$parameter where name like '%size';`
  - za prikaz bafera pojedinačno
  - `select name, value from v$parameter where name like 'java_pool_size';`
  - `select name, value from v$parameter where name='sga_target';`
- Ako je java pool na 0, a SGA ima vrijednost, onda to znači da je Automatic Storage Management (ASM) uključen. Ako stavimo SGA na 0, onda podešavamo svaki bafer ručno.
  - `alter system set sga_target=0;`

# Database Schema Objects and Managing Storage

---

# Database Schema Objects and Managing Storage

- Procjena veličine tabele preko veličine segmenata (npr. korisnik.radnik)
- Logovati se kao sys u PDB putem SQL Developer- a
  - with tablesize as (

```
select segment_name,segment_type, sum(bytes/1024/1024) MB
from dba_segments
where segment_name='RADNIK'
group by segment_name,segment_type
)
```

```
select count(*), mb current_table_size_mb, round(mb/count(*), 2)
row_size_mb, round(mb/count(*)*1000, 2) estimated_size_mb
from tablesize, korisnik.radnik
group by mb;
```

# Database Schema Objects and Managing Storage

- Da li je sprovedena procjena dovoljno precizna? Mane?
- Zadatak 1
  - Pogledati na koji način može da se uradi reorganizacija tabele.  
Pogledati šta je Segment Advisor

# Managing Resources

---

# Managing Resources

- Resource plan određuje kako su resursi raspoređeni među grupama korisnika
- Resource grupa korisnika je skup korisnika koji imaju slične zahtjeve za upotrebom sistema
- Kreirati consumer grupu u okviru koje se CPU raspoređuje po round robin algoritmu. Dodati korisnika "korisnik" u kreiranu grupu.

# Managing Resources

- U oracle 21c potrebno je prvo omogućiti kreiranje pripremnog područja u okviru kog se skladište informacije o novokreiranoj consumer grupi

- BEGIN

```
    DBMS_RESOURCE_MANAGER.CREATE_PENDING_AREA();
```

```
END;
```

- Kreiranje grupe

- BEGIN

```
    DBMS_RESOURCE_MANAGER.CREATE_CONSUMER_GROUP(
```

```
        consumer_group => 'OLTP',
```

```
        mgmt_mth        => 'ROUND-ROBIN',
```

```
        comment         => 'OLTP applications'
```

```
    );
```

```
END;
```

# Managing Resources

- Dodavanje korisnika u grupu

- BEGIN

```
DBMS_RESOURCE_MANAGER.SET_CONSUMER_GROUP_MAPPING(  
    ATTRIBUTE      => DBMS_RESOURCE_MANAGER.ORACLE_USER,  
    VALUE          => 'KORISNIK',  
    CONSUMER_GROUP => 'OLTP'  
);  
END;
```

- Pregled postojećih grupa

- `select *`  
`from dba_rsrc_consumer_groups;`

# Managing Resources

- Zadatak 1
  - Pogledati kako se dodaje consumer grupa u neki resource plan, pregledati šta su leveli i kako se zadaju.

# Automatic Tasks With the Scheduler

---

# Automatic Tasks With the Scheduler

- Kreirati Job (posao) u okviru kog će se kreirati tabela korisnik.projekat
  - nazp varchar2(20)
  - opis varchar2(50)

# Automatic Tasks With the Scheduler

- Logovati se kao sys u PDB putem SQL Developer- a

- BEGIN

```
DBMS_SCHEDULER.CREATE_JOB (  
  job_name          => 'KREIRANJE_TABELE',  
  job_type          => 'PLSQL_BLOCK',  
  job_action        => 'BEGIN EXECUTE IMMEDIATE ''CREATE TABLE  
KORISNIK.PROJEKAT(NAZP VARCHAR2(20), OPIS VARCHAR2(50))''; END;',  
  start_date        => SYSTIMESTAMP,  
  enabled            => TRUE,  
  auto_drop         => TRUE,  
  comments          => 'Posao za kreiranje tabele');  
END;
```

# Automatic Tasks With the Scheduler

- Prikaz svih poslova
  - `select job_name, enabled, state, next_run_date, repeat_interval  
from dba_scheduler_jobs;`
- Uklanjanje posla
  - `begin  
    dbms_scheduler.drop_job(job_name => 'KREIRANJE_TABELE');  
end;`

# Automatic Tasks With the Scheduler

- Parametri
  - JOB\_TYPE
    - PLSQL\_BLOCK
    - STORED\_PROCEDURE
    - EXECUTABLE
    - CHAIN
    - SQL\_SCRIPT
    - BACKUP\_SCRIPT
    - EXTERNAL\_SCRIPT

# Automatic Tasks With the Scheduler

- Kreirati program (Program) za dodavanje jednog reda u tabelu korisnik.radnik
- Logovati se kao sys u PDB putem SQL Developer- a

- BEGIN

```
DBMS_SCHEDULER.CREATE_PROGRAM (  
  program_name          => 'ins_radnik_slog',  
  program_type          => 'PLSQL_BLOCK',  
  program_action        => 'begin execute immediate ''insert into  
korisnik.radnik values(korisnik.seq_mbr.nextval, ''A'', ''B'', 50000,  
''Z'')''; execute immediate ''commit''; end;',  
  enabled               => TRUE,  
  comments              => 'Dodavanje sloga u transakciji');  
END;
```

# Automatic Tasks With the Scheduler

- Prikaz svih programa

- `select program_name, program_type, enabled  
from dba_scheduler_programs;`

- Uklanjanje programa

- `BEGIN`

- `dbms_scheduler.drop_program(program_name => 'ins_radnik_slog');`
  - `END;`

# Automatic Tasks With the Scheduler

- Zadatak 1
  - Kreirati posao (Job) kojim će se tokom 2 minuta na svakih 10 sekundi pokretati prethodno kreirani program (ins\_radnik\_slog)!
  - Prilikom kreiranja job- a ne postavljati da je enabled nego to odraditi nakon što se kreira posao

# Automatic Tasks With the Scheduler

- Kreirati zakazivač koji će od sadašnjeg trenutka narednih 30 dana da se aktivira (okida) jednom dnevno
- Logovati se kao sys u PDB putem SQL Developer- a

- BEGIN

```
DBMS_SCHEDULER.CREATE_SCHEDULE (  
    schedule_name      => 'mjesecni_zakacivac',  
    start_date         => SYSTIMESTAMP,  
    end_date           => SYSTIMESTAMP + INTERVAL '30' day,  
    repeat_interval    => 'FREQ=DAILY; INTERVAL=1',  
    comments           => 'Every day');
```

```
END;
```

# Automatic Tasks With the Scheduler

- Prikaz svih zakazivača

- `select *`  
`from dba_scheduler_schedules;`

- Uklanjanje zakazivača

- `BEGIN`  
`DBMS_SCHEDULER.DROP_SCHEDULE(schedule_name => 'mjesecni_zakacivac');`  
`END;`

# Automatic Tasks With the Scheduler

- Zadatak 2
  - Prilagoditi zadatak 1 tako da se koristi prethodno napravljeni zakazivač (mjesečni zakazivač) umjesto onoga što je zadato u zadatku 1

# Kraj!

Hvala na pažnji!