

## IDŹ DO

PRZYKŁADOWY ROZDZIAŁ



SPIS TREŚCI

## KATALOG KSIĄŻEK

KATALOG ONLINE

ZAMÓW DRUKOWANY KATALOG

## TWÓJ KOSZYK

DODAJ DO KOSZYKA

## CENNIK I INFORMACJE

ZAMÓW INFORMACJE  
O NOWOŚCIACH

ZAMÓW CENNIK

## CZYTELNIA

FRAGMENTY KSIĄŻEK ONLINE

# Oracle RMAN.

## Leksykon kieszonkowy

Autorzy: Darl Kubn, Scott Schulze

Tłumaczenie: Bartłomiej Garbacz

ISBN: 83-7197-673-9

Tytuł oryginału: [Oracle RMAN. Pocket Reference](#)

Format: B6, stron: 152



Niniejsza książka jest leksykonem kieszonkowym pakietu Recovery Manager (RMAN) – narzędzia służącego do przeprowadzania wszelkich czynności związanych z archiwizowaniem i odzyskiwaniem danych w bazach danych Oracle. Nie stanowi ona jednak wyczerpującego kompendium wiedzy na temat archiwizowania i odzyskiwania danych. Zawarto tu natomiast przegląd architektury pakietu RMAN, w skrócie opisano metody archiwizowania i odzyskiwania danych za pomocą tego pakietu, przedstawiono zagadnienia związane z obsługą katalogu odzyskiwania (recovery catalog), a także zamieszczono diagramy składniowe poleceń pakietu RMAN.

Zadaniem niniejszego Leksykonu jest umożliwienie szybkiego odnalezienia opisu składni i odpowiedniego przykładu zastosowania potrzebnego polecenia pakietu RMAN w celu archiwizacji, odtworzenia lub odzyskania danych. Autorzy założyli, że Czytelnik jest zaznajomiony z podstawami obsługi baz danych Oracle oraz z zagadnieniami związanymi z tworzeniem kopii zapasowej i odzyskiwaniem danych. Wszystkie zamieszczone w niniejszej książce przykłady skryptów trybu wsadowego zostały utworzone w języku skryptowym systemu Unix. Wiele z tych przykładów zawiera ścieżki dostępu odpowiadające środowisku pracy stosowanemu przez Autorów. Jeśli Czytelnik będzie opracowywał własne skrypty, musi je dostosować do środowiska pracy swojej aplikacji.

Pisząc niniejszą książkę Autorzy starali się stosować do prawa Pareto dlatego położono szczególny nacisk na opisanie tych zagadnień, z którymi użytkownik najprawdopodobniej zetknie się podczas wykorzystywania pakietu RMAN. Nie omówiono tu wszystkich rodzajów środowiska pracy ani wszystkich scenariuszy archiwizowania i odzyskiwania danych, jakie stosować może administrator baz danych Oracle.

W niniejszej książce opisano niektóre z najczęściej spotykanych scenariuszy, jednakże trzeba pamiętać, że sprawą o fundamentalnym znaczeniu jest dogłębne poznanie własnej implementacji pakietu RMAN. Jest to warunek zapewnienia sobie możliwości odzyskania każdej bazy danych, bez względu na rodzaj ewentualnej awarii. Nie można przecenić ważności regularnego testowania pakietu jako formy przygotowania do usuwania skutków nieprzewidywalnych awarii.

Dzięki poprawnie skonfigurowanej implementacji pakietu oraz jego regularnemu testowaniu administrator bazy danych zyskuje pewność, że niezależnie od okoliczności zdoła on opanować sytuację.



## *Spis treści*

Wstęp.....	7
Architektura pakietu RMAN.....	10
Program wykonywalny RMAN.....	11
Procesy serwera .....	12
Kanały.....	12
Docelowa baza danych .....	13
Baza danych katalogu odzyskiwania .....	13
Warstwa obsługi nośników danych .....	16
Kopie zapasowe, zbiory kopii zapasowych i elementy kopii zapasowych.....	17
Uruchamianie pakietu RMAN.....	18
Wymagania dotyczące docelowej bazy danych .....	19
Wywoływanie programu wykonywalnego pakietu RMAN..	23
Łączenie z docelową bazą danych bez wykorzystania bazy danych katalogu odzyskiwania ...	24
Łączenie z docelową bazą danych oraz bazą danych katalogu odzyskiwania.....	27
Wykonywanie poleceń .....	29
Wykonywanie komend wywoływanych z wiersza poleceń RMAN.....	29
Wykonywanie komend pakietu RMAN zapisanych w pliku .....	32
Wykonywanie poleceń języka SQL i systemu operacyjnego z poziomu programu RMAN .....	34
Uruchamianie programu RMAN z poziomu skryptów systemowych.....	35

Wykorzystywanie katalogu odzyskiwania .....	39
Istota katalogu odzyskiwania.....	39
Strategie tworzenia kopii zapasowej bazy katalogu .....	43
Zagadnienia związane z katalogiem odzyskiwania .....	46
Tworzenie katalogu .....	49
Rejestrowanie docelowej bazy danych.....	52
Skrypty składowane w katalogu.....	53
Przechowywanie skryptu .....	53
Uruchamianie skryptu składowanego.....	54
Przeglądanie skryptu składowanego.....	56
Usuwanie skryptu składowanego .....	57
Kopie zapasowe.....	58
Pełna kopia zapasowa zamkniętych plików danych.....	58
Pełna kopia zapasowa otwartych plików danych .....	60
Tworzenie kopii zapasowej przestrzeni tabel.....	61
Tworzenie kopii zapasowej plików danych.....	61
Tworzenie kopii zapasowej plików kontrolnych.....	62
Tworzenie kopii zapasowej archiwalnych plików dziennika powtórzeń .....	64
Przyrostowe kopie zapasowe .....	66
Porady dotyczące tworzenia kopii zapasowych .....	70
Odtwarzanie plików .....	73
Odtwarzanie i odzyskiwanie wszystkich plików danych ...	74
Odtwarzanie konkretnych przestrzeni tabel lub plików danych.....	76
Odtwarzanie przestrzeni tabel zapisanych w trybie tylko do odczytu .....	77
Odtwarzanie do określonego momentu w czasie .....	78
Odtwarzanie plików kontrolnych .....	79
Odtwarzanie archiwalnych plików dziennika powtórzeń...	80

Lista poleceń pakietu RMAN.....	81
Powtarzające się słowa kluczowe.....	81
Powtarzające się wyrażenia.....	84
Składnia poleceń wywoływanych z wiersza poleceń.....	84
@.....	87
@@.....	87
ALLOCATE CHANNEL.....	88
ALLOCATE CHANNEL FOR MAINTENANCE.....	90
ALTER DATABASE.....	90
BACKUP.....	91
BLOCKRECOVER (dotyczy tylko Oracle9i).....	97
CATALOG.....	98
CHANGE.....	100
CONFIGURE (dotyczy tylko Oracle9i).....	102
CONNECT.....	105
COPY.....	106
CREATE CATALOG.....	108
CREATE SCRIPT.....	109
CROSSCHECK.....	110
DELETE.....	111
DELETE SCRIPT.....	114
DROP CATALOG.....	114
DUPLICATE.....	115
EXECUTE SCRIPT.....	117
EXIT.....	117
HOST.....	117
LIST.....	118
listaObiektów.....	121
opcjeKonserwacji (dotyczy tylko Oracle9i).....	121
opcjeZachowywania (dotyczy tylko Oracle9i).....	122
operandPrzydziału.....	123
PRINT SCRIPT.....	125
przestKopieBezp (dotyczy tylko Oracle9i).....	125

QUIT (dotyczy tylko Oracle9i) .....	126
RECOVER.....	126
REGISTER .....	128
RELEASE CHANNEL.....	129
releasePoKonservacji.....	129
REPLACE SCRIPT .....	130
REPLICATE .....	132
REPORT .....	132
RESET DATABASE.....	135
RESTORE.....	135
RESYNC.....	138
RUN.....	138
SEND .....	139
SET .....	139
SHOW (dotyczy tylko Oracle9i) .....	141
SHUTDOWN .....	142
specArchDzienPowt .....	142
specCzasuUtworz .....	144
specŁańcuchaPołączenia .....	144
specObiektówKonservacji (dotyczy tylko Oracle9i).....	145
specPlikuDanych .....	146
specUrządzenia .....	146
SPOOL (dotyczy tylko Oracle9i) .....	147
SQL.....	147
STARTUP.....	148
SWITCH .....	148
UPGRADE CATALOG .....	150
VALIDATE .....	150
wyrażenieUntil.....	151

## *Kopie zapasowe*

W niniejszym rozdziale zaprezentowano serię skryptów pakietu RMAN oraz skryptów systemowych (*korn*), które służą do tworzenia kopii zapasowych bazy danych oraz jej poszczególnych komponentów. Pakiet RMAN może archiwizować pliki danych (*datafiles*), pliki kontrolne (*control files*), archiwalne pliki dziennika powtórzeń (*archived redo files*) oraz elementy kopii zapasowej (*backup pieces*). Pakiet RMAN nie tworzy kopii zapasowych aktywnych plików dziennika powtórzeń (*online redo logs*).

---

### *UWAGA*

Częścią standardowego działania pakietu RMAN w odniesieniu do archiwizacji danych jest odczytywanie wszystkich bloków plików danych, sprawdzenie ich poprawności, a w końcu zapisanie informacji o wszelkich wykrytych nieprawidłowościach. Informacje dotyczące potencjalnych uszkodzeń są dostępne w widokach `V$BACKUP_CORRUPTION` oraz `V$COPY_CORRUPTION`.

---

Przed utworzeniem kopii zapasowej bazy danych musi być ona zamontowana (tryb *mount*) lub otwarta. Wymóg, by baza danych była przynajmniej w trybie *mount*, jest spowodowany koniecznością zapewnienia dostępu programowi RMAN do plików kontrolnych docelowej bazy danych przed przeprowadzeniem procesu archiwizacji danych.

## *Pełna kopia zapasowa zamkniętych plików danych*

Przed utworzeniem kopii zapasowej zamkniętych plików danych (*offline backup*) należy zamknąć bazę danych, a następnie uruchomić ją ponownie w trybie *mount*. Baza danych nie musi znajdować się w trybie *archivelog*.

Poniższy skrypt (dla wersji Oracle9i\*) zamyka bazę danych, montuje ją (tryb *mount*), tworzy jej kopię zapasową, a w końcu ponownie ją otwiera:

```
#!/bin/ksh

rman target / <<EOF
shutdown immediate;
startup mount;
backup database format
  '/d99/rmanback/brdstn/rman_%d_%t_%U.bus';
alter database open;
EOF
exit
```

Poniższy skrypt systemowy dla wersji Oracle8i tworzy pełną kopię zapasową bazy danych. Należy zwrócić uwagę, że składnia poleceń jest tu nieco bardziej złożona w porównaniu z wersją Oracle9i. Dla wersji Oracle8i polecenia `allocate` oraz `backup` muszą znajdować się wewnątrz polecenia `run{ }`.

```
#!/bin/ksh

rman target / nocatalog <<EOF
shutdown immediate;
startup mount;
run {
allocate channel d1 type disk;
backup database format
  '/d99/rmanback/brdstn/rman_%d_%t_%U.bus';
}
alter database open;
EOF
Exit
```

---

\* Wyznacznikiem nie jest wersja serwera Oracle, ale programu *Recovery Manager*. Np. dla wydania RMAN 8.1.7.0.0 z serwerem w wersji 9i należy zastosować wersję skryptu dla Oracle8i — *przyp. red.*

---

### UWAGA

W obu przykładach zastosowano parametr *format* w celu podania określonej lokalizacji oraz niepowtarzalnej nazwy dla elementów kopii zapasowej. Pominięcie tego parametru powoduje, że elementy kopii zapasowej kierowane są do katalogu *\$ORACLE\_HOME/dbs*. Więcej szczegółów na temat tego zagadnienia znajduje się w tabeli 1.1 w rozdziale „Lista poleceń pakietu RMAN”.

---

## *Pełna kopia zapasowa otwartych plików danych*

Przed utworzeniem kopii zapasowej otwartych plików danych (*online backup*) baza danych musi być otwarta oraz znajdować się w trybie *archivelog*. Jeśli wykorzystywanym systemem jest Oracle9i, składnia odpowiedniego polecenia jest bardzo prosta:

```
RMAN> backup database format  
2> '/d99/rmanback/brdstn/rman_%d_%t_%U.bus' ;
```

---

### UWAGA

W przeciwieństwie do standardowego sposobu tworzenia kopii zapasowych otwartych plików danych (gorących kopii zapasowych) pakiet RMAN nie przełącza przestrzeni tabel w tryb *backup*. Nie są zatem generowane żadne dodatkowe dzienniki powtórzeń. W przypadku baz danych o wysokiej aktywności może to dawać znaczące korzyści związane z lepszym wykorzystaniem zasobów systemowych.

---

W przypadku wersji Oracle8i należy zastosować polecenie `run {`:

```
RMAN> run {  
2> allocate channel d1 type disk;
```

```
3> backup database format
4> '/d99/rmanback/brdstn/rman_%d_%t_%U.bus';
5> }
```

## *Tworzenie kopii zapasowej przestrzeni tabel*

Możliwość podawania jedynie podzbioru przestrzeni tabel podczas procesu archiwizacji danych zwiększa zakres możliwych strategii tworzenia kopii zapasowych. W przypadku wersji Oracle9i składnia polecenia jest bardzo prosta:

```
RMAN> backup tablespace system, users format
2> '/d99/rmanback/brdstn/rman_bckup_%d_%t_%U.bus';
```

W przypadku wersji Oracle8i należy ponownie zastosować polecenie `run{}`:

```
RMAN> run {
2> allocate channel d1 type disk;
3> backup format
4> '/d99/rmanback/brdstn/rman_bckup_%d_%t_%U.bus'
5> tablespace system, users include current
   controlfile;
6> }
```

## *Tworzenie kopii zapasowej plików danych*

Pliki danych można archiwizować przez podanie ich numerów lub nazw. Dla wersji Oracle9i składnia polecenia jest prosta. W przypadku tworzenia kopii zapasowej na dysku i jeśli numery plików archiwizowanych danych są znane, stosuje się następujące polecenie:

```
RMAN> backup datafile 1,2;
```

W analogicznej sytuacji dla wersji Oracle8i składnia takiego polecenia wygląda następująco:

```
RMAN> run {
2> allocate channel d1 type disk;
3> backup format
4> '/d99/rmanback/brdstn/rman_bckup_%d_%t_%U.bus'
5> datafile 1, 2;
6> }
```

Można także podać nazwę konkretnego pliku danych przeznaczonych do archiwizacji:

```
RMAN> run {
2> allocate channel d1 type disk;
3> backup format
4> '/d99/rmanback/brdstn/rman_bckup_%d_%t_%U.bus'
5> '/d01/oradata/workprd/users_01.dbf';
6> }
```

Jeżeli archiwizacji poddaje się więcej plików, ich nazwy należy oddzielić przecinkami.

## *Tworzenie kopii zapasowej plików kontrolnych*

Przydatną właściwością systemu Oracle9i jest możliwość automatycznej archiwizacji plików kontrolnych po każdorazowym wydaniu polecenia backup lub copy. Można to osiągnąć za pomocą polecenia configure w następujący sposób:

```
RMAN> configure controlfile autobackup on;
```

Samodzielnie archiwizację pliku kontrolnego w systemie Oracle9i przeprowadza się za pomocą polecenia:

```
RMAN> backup current controlfile;
```

Poniżej przedstawiono przykład archiwizacji pliku kontrolnego za pomocą skryptu systemowego. Jest on poprawny dla wersji Oracle9i i Oracle8i:

```
#!/bin/ksh
# Plik: kopia_bezp_kontrol.ksh
#-----
export ORACLE_HOME=/d00/app/oracle/product/9.0.0
export ORACLE_SID=brdstn
export PATH=/usr/sbin:/usr/bin:$ORACLE_HOME/bin
export MAILX="/usr/ucb/Mail"
export MAIL_LIST="chaya@garam.com"
#-----
BOX=`uname -a | awk '{print$2}'`
#-----
date
#-----
rman nolog log=rman_kopia_kontrol.log <<EOF
connect target /
run {
  allocate channel d1 type disk;
  setlimit channel d1 kbytes 1900000;
  backup
    format '/d99/rmanback/brdstn/
    rman_cntl_%d_%t_%U.bct'
    (current controlfile);
  release channel d1;
}
EOF
#
if [ $? -ne 0 ]; then
$MAILX -s "RMAN napotkał problem z $ORACLE_SID
w $BOX" \
$MAIL_LIST <<EOF
Sprawdź kopię bezpieczeństwa RMAN pliku
kontrolnego...
EOF
#
else
print "Kopia bezpieczeństwa RMAN pliku
kontrolnego została poprawnie utworzona..."
fi
```

```
#-----  
date  
#-----
```

## *Tworzenie kopii zapasowej archiwalnych plików dziennika powtórzeń*

W przypadku wersji Oracle9i składnia polecenia tworzącego kopię zapasową archiwalnych plików dziennika powtórzeń (*archived redo log files*) jest bardzo prosta:

```
RMAN> backup archivelog all;
```

Inną techniką, możliwą do zastosowania w systemie Oracle9i, jest zastosowanie w poleceniu `backup` wyrażenia *plus archivelog*, dzięki czemu archiwalne pliki dziennika powtórzeń wchodziły w skład kopii zapasowej. Wydanie poniższej instrukcji powoduje utworzenie co najmniej dwóch elementów kopii zapasowej — dla plików danych i dla archiwalnych plików dziennika powtórzeń:

```
RMAN> backup database format  
2> '/d00/backup/rman_%U.bus' plus archivelog;
```

Znajdujący się poniżej prosty skrypt działa poprawnie zarówno w przypadku wersji Oracle8i, jak i Oracle9i. Wykonanie skryptu powoduje utworzenie kopii zapasowej wszystkich archiwalnych plików dziennika powtórzeń docelowej bazy danych:

```
RMAN> run {  
2> allocate channel t1 type 'sbt_tape';  
3> backup filesperset 5 format  
4> '/d0100/backup/ar_%d_%t.bus' (archivelog all );  
5> }
```

---

## UWAGA

Przydatną właściwością pakietu RMAN jest jego zdolność odróżniania tych archiwalnych plików dziennika powtórzeń, których zapis na dysku jest zakończony. Dzięki temu pakiet RMAN nigdy nie podejmuje próby archiwizacji częściowo zapisanych archiwalnych plików dziennika powtórzeń.

---

Poniżej przedstawiono kolejny przykład skryptu systemowego, który za pomocą poleceń pakietu RMAN tworzy kopię zapasową archiwalnych plików dziennika powtórzeń. Po jego uruchomieniu, w zbiorze kopii zapasowej znajdują się wszystkie archiwalne pliki dziennika powtórzeń utworzone w ciągu ostatnich trzech dni.

```
#!/bin/ksh
# Plik: kopia_bezp_arch.ksh
#-----
export ORACLE_HOME=/d00/app/oracle/product/9.0.0
export ORACLE_SID=brdstn
export PATH=/usr/sbin:/usr/bin:$ORACLE_HOME/bin
export MAILX="/usr/ucb/Mail"
export MAIL_LIST="heeraa@chamak.com"
#-----
BOX=`uname -a | awk '{print$2}'`
#-----
date
#-----
rman nocatalog log=rman_kopia_arch.log <<EOF
connect target sys/pwd
run {
  allocate channel d1 type disk;
  setlimit channel d1 kbytes 1900000;
  backup
    format '/d99/backup/brdstn/arch_%d_%t.bus'
    filesperset=50
    archivelog from time 'sysdate-3';
  release channel d1;
}
```

```

EOF
#
if [ $? -ne 0 ]; then
$MAILX -s "RMAN napotkał problem z $ORACLE_SID
w $BOX" \
$MAIL_LIST <<EOF
Sprawdź kopię bezpieczeństwa RMAN archiwalnego
dziennika powtórzeń...
EOF
#
else
print "Kopia bezpieczeństwa RMAN archiwalnego
dziennika powtórzeń została poprawnie
utworzona..."
fi
#-----
date
#-----

```

---

### UWAGA

Jeżeli jakiś proces nie związany z programem RMAN usunie pliki dziennika powtórzeń, to ani w plikach kontrolnych, ani w opcjonalnym katalogu odzyskiwania nie zostanie zapisana żadna informacja o tym, że te pliki nie są już dostępne. Dlatego też należy okresowo wydawać polecenie:

```
RMAN> change archivelog all validate;
```

Z tego względu warto, aby program RMAN usuwał te pliki z systemu w momencie, gdy nie są one już potrzebne.

---

## *Przyrostowe kopie zapasowe*

Jedną z najbardziej spektakularnych cech pakietu RMAN jest możliwość tworzenia przyrostowych kopii zapasowych (*incremental backups*). W przypadku standardowej, opartej na powielaniu plików, procedury tworzenia kopii zapasowych (na przykład

tworzenia gorących lub zimnych kopii zapasowych) kopiowane są wszystkie bloki plików danych bazy docelowej, bez względu na to, czy były one używane, czy nie. Program RMAN posiada możliwość sprawdzenia, które bloki plików danych zostały zmienione od ostatniej archiwizacji. Dzięki temu kopiowaniu podlegają tylko bloki zmodyfikowane.

---

### UWAGA

Jeśli podczas archiwizacji program RMAN kopiuje tylko zmodyfikowane bloki, proces ten nazywa się kompresją (*compression*). Nieużywane lub niezmodyfikowane bloki w pliku danych zostają pominięte.

---

Pomijanie nie zmienionych bloków daje ogromne korzyści poprzez zmniejszenie wymagań co do zasobów systemowych. Oznacza to także, że czas wykonania kopii zapasowej oraz ilość potrzebnego do jej przechowania miejsca na taśmie lub dysku zależą od ilości dokonanych w bazie danych zmian, nie zaś od jej rozmiaru. W przypadku bardzo dużych baz danych już tylko ten jeden argument może uczynić zasadnym korzystanie z pakietu RMAN.

Wiele zagadnień dotyczących kopii przyrostowych związanych jest z pojęciem poziomu (*level*). Podstawą idei tworzenia przez pakiet RMAN wielopoziomowych kopii zapasowych jest założenie, że archiwizować należy tylko te bloki, które zostały zmodyfikowane od czasu poprzedniej archiwizacji. Tworzenie kopii przyrostowych można stosować w odniesieniu do całej bazy danych, jak również danych przestrzeni tabel lub plików danych. Posługiwanie się poziomami określone jest w przypadku systemów Oracle jako tworzenie wielopoziomowej przyrostowej kopii zapasowej (*multilevel incremental backup*). Istnieje pięć poziomów kopii zapasowej w przedziale od 0 do 4 (*level 0 – level 4*).

Istnieją dwa rodzaje przyrostowych kopii zapasowych — różnicowe (*differential*) oraz kumulacyjne (*cumulative*). Tworzenie różnicowej kopii przyrostowej polega na tym, że pakiet RMAN archiwizuje tylko te bloki, które zostały zmienione od czasu utworzenia kopii poziomu  $n$  lub niższego. Przykładowo, jeśli tworzy się różnicową kopię przyrostową poziomu 1, archiwizacji podlegają bloki, które zostały zmodyfikowane od czasu poprzedniej archiwizacji na poziomie 1. W przypadku pakietu RMAN tworzenie kopii różnicowych jest domyślnym trybem przeprowadzania archiwizacji przyrostowej.

---

#### UWAGA

W razie tworzenia przyrostowej kopii zapasowej o poziomie wyższym od 0, jeśli nie wykonano wcześniej żadnej archiwizacji na tym poziomie, pakiet RMAN automatycznie tworzy kopię poziomu 0.

---

Tworzenie kumulacyjnej kopii przyrostowej polega na tym, że pakiet RMAN archiwizuje tylko te bloki, które zostały zmienione od czasu utworzenia kopii poziomu  $n-1$  lub niższego. Przykładowo, jeśli tworzy się kumulacyjną kopię przyrostową poziomu 1, nastąpi archiwizacja bloków, które zostały zmodyfikowane od czasu ostatniej archiwizacji na poziomie 0.

Udostępnienie tych wszystkich możliwości pozwala na wybór najbardziej odpowiedniej w danej sytuacji strategii archiwizacji danych. Różnicowa kopia bezpieczeństwa zajmuje mniej miejsca i jej utworzenie trwa krócej, ale odtworzenie danych na podstawie tej kopii zabiera więcej czasu. Z kolei kopia kumulacyjna zajmuje więcej miejsca i więcej czasu trwa jej utworzenie, ale odtworzenie danych na jej podstawie jest szybsze. Zatem w zależności od specyficznych cech aplikacji należy podjąć decyzję, czy ważniejsze jest zminimalizowanie czasu tworzenia kopii

zapasowych danych, czy też czasu odtwarzania danych. W przypadku małych baz danych zaleca się wykorzystanie pakietu RMAN do codziennego wykonywania kopii zapasowej poziomu 0.

---

### UWAGA

Pełna archiwizacja (*full backup*) polega na tworzeniu kopii zapasowej dokładnie tych samych bloków, co w przypadku archiwizacji na poziomie 0. Różnica między tymi procesami wynika z tego, że kolejne procesy archiwizacji przyrostowej zachodzą w sposób całkowicie niezależny od istnienia pełnej kopii bezpieczeństwa. Dlatego też pełne kopie nie mogą zostać zastosowane w celu odtwarzania danych z kopii przyrostowych. Tworzenie pełnej kopii zapasowej jest domyślnym trybem archiwizacji w przypadku, gdy nie podano żadnego poziomu przyrostu.

---

Poniżej znajduje się przykład utworzenia kopii kumulacyjnej (dla systemu Oracle9i), gdzie program RMAN archiwizuje wszystkie bloki, które zostały zmodyfikowane od czasu ostatniej archiwizacji na poziomie 3 lub niższym, oraz pomija przeszerzenie tabel przeznaczone tylko do odczytu (*read only*):

```
RMAN> backup incremental level 4 cumulative database
2> skip readonly;
```

Poniżej przedstawiono przykład utworzenia różnicowej kopii bezpieczeństwa w przypadku systemu Oracle8i. Archiwizacji podlegają wszystkie bloki, które zostały zmodyfikowane od czasu ostatniej archiwizacji na poziomie 1 lub niższym:

```
RMAN> run {allocate channel d1 type disk;
2> backup incremental level 1 format
3> '/d0101/backup/rman_%U.bus' database;}
```

## *Porady dotyczące tworzenia kopii zapasowych*

Powyżej omówiono różne scenariusze archiwizacji danych. W dalszej części niniejszego rozdziału przedstawiono polecenia służące do informowania o spójności kopii zapasowych lub zwiększania wydajności procesu ich tworzenia.

### *Sprawdzanie stanu kopii zapasowej*

Jedną z użytecznych cech pakietu RMAN jest możliwość kontrolowania stanu kopii zapasowej. Najprostszym sposobem zapoznania się z tymi informacjami jest wydanie polecenia `list backup`.

```
RMAN> list backup;
```

Efektom wykonania polecenia `list backup` jest wyświetlenie informacji o nazwie, poziomie, czasie utworzenia oraz stanie każdej z kopii zapasowych. Ponadto pakiet RMAN ma możliwość prostego sprawdzenia odtwarzalności plików danej kopii zapasowej. Do tego celu służy polecenie `restore database validate`. Efektom wykonania tej komendy nie jest przeprowadzenie samego procesu odtwarzania, lecz jedynie sprawdzenie, czy zawartość zbiorów kopii zapasowej może zostać w razie potrzeby odtworzona. Poniżej przedstawiono przykład zastosowania tego polecenia dla aplikacji działającej w systemie Oracle8i i Oracle9i:

```
RMAN> run {  
2> allocate channel d1 type disk;  
3> restore database validate;  
4> }
```

## *Konfigurowanie kanałów*

Przydatną cechą systemu Oracle9i jest możliwość zmiany domyślnych właściwości kanału. Po zmianie właściwości te zostają zapisane w pliku kontrolnym i są zachowywane na potrzeby kolejnych sesji programu RMAN. Poniżej przedstawiono przykład zmiany domyślnej wartości parametru *rate* (maksymalna liczba danych czytanych przez kanał) urządzenia dyskowego oraz wartości łańcucha formatującego:

```
RMAN> configure channel device type disk  
2> rate 1000000 format '/d99/rmanback/rman_%U.bus';
```

Przeglądanie wszystkich ustawień konfiguracyjnych jest możliwe po wydaniu dostępnego w wersji Oracle 9i polecenia `show all`:

```
RMAN> show all;
```

RMAN configuration parameters are:  
CONFIGURE RETENTION POLICY TO REDUNDANCY 1;  
# default  
CONFIGURE BACKUP OPTIMIZATION OFF; # default  
CONFIGURE DEFAULT DEVICE TYPE TO DISK; # default  
CONFIGURE CONTROLFILE AUTOBACKUP OFF; # default  
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE  
TYPE DISK TO '%F'; # default  
CONFIGURE DEVICE TYPE DISK PARALLELISM 1;  
# default  
CONFIGURE DATAFILE BACKUP COPIES FOR DEVICE TYPE  
DISK TO 1; # default  
CONFIGURE ARCHIVELOG BACKUP COPIES FOR DEVICE TYPE  
DISK TO 1; # default  
CONFIGURE CHANNEL DEVICE TYPE DISK RATE 1000000  
FORMAT '/d99/rmanback/rman\_%U.bus';  
CONFIGURE MAXSETSIZE TO UNLIMITED; # default  
CONFIGURE SNAPSHOT CONTROLFILE NAME TO '/d00/  
app/oracle product/9.0.0/dbs/snapcf\_dev900.f';  
# default

Warto zauważyć, że pozycje, dla których ustawienia domyślne nie zostały zmienione, są oznaczone przez # default.

Aby skasować ustawienia właściwości kanału, wydaje się polecenie `configure clear`. Poniżej przedstawiono przykład polecenia, którego wydanie przywraca domyślne ustawienia właściwości urządzenia:

```
RMAN> configure default device type clear;
```

### *Równoległość tworzenia kopii zapasowych*

W celu zwiększenia wydajności pracy aplikacji program RMAN może rozdzielić wykonanie pewnych poleceń na kilka równoległych procesów. Nie oznacza to, że w razie wywołania kilku poleceń będą one wszystkie wykonywane równocześnie, ale że wykonanie każdego z nich oddzielnie zostanie rozbite na równoległe procesy.

Dobrym przykładem przetwarzania równoległego (*parallelization*) jest wykonanie polecenia `backup`. Jeśli przydzielonych jest kilka kanałów, program RMAN jednocześnie korzysta z kilku procesów w celu wykonania tego polecenia. Dane zapisywane są jednocześnie przez wszystkie kanały:

```
RMAN> configure device type disk parallelism 3;  
RMAN> backup database format '/d99/rmanback/  
rman_%U.bus';
```

W efekcie podczas archiwizacji bazy danych wykorzystywane są trzy równoległe procesy.

## Oznaczanie kopii zapasowych

Pakiet RMAN udostępnia możliwość nadawania zbiorom kopii zapasowych nazw odpowiadających wymaganiom użytkownika. Przykładowo, pozwala to na łatwe rozróżnianie kopii zapasowych tworzonych codziennie i co tydzień, lub też na rozróżnianie przyrostowych kopii zapasowych tworzonych na różnych poziomach. W celu nadania odpowiedniej nazwy stosuje się słowo kluczowe *tag* w poleceniu *backup*. Poniżej przedstawiono przykład oznaczenia kopii zapasowej poziomu 3 jako *inc\_level\_3*:

```
RMAN> backup incremental level 3 tag inc_level_3
      format
2> '/d99/rmanback/rman_%U.bus';
```

Znaczniki (*tags*) mogą być stosowane w poleceniu *list* w celu szybkiego zlokalizowania zbiorów kopii zapasowych, a także w poleceniach *restore* oraz *switch*. Ta sama wartość znacznika może być wykorzystana dla wielu różnych zbiorów kopii zapasowych, zatem nie musi być ona niepowtarzalna. Jeśli wartość danego znacznika jest niejednoznaczna, zostanie ona zastosowana do opisanego ostatnio utworzonego zbioru kopii zapasowych o podanej w znaczniku wartości.

W poniższym przykładzie przedstawiono zastosowanie polecenia *list* wyświetlającego informacje o istniejącej kopii zapasowej, którą oznaczono jako *inc\_level\_3*:

```
RMAN> list backuppiece tag=inc_level_3;
```