

Seab@er Software AG

Oracle Dokumentation 05 - Backup & Restore



1.	Vorwort	6
2.	Rman	7
2.1	Rman starten	7
2.2	Konfiguration	8
2.2.1	Einstellungen Anzeigen	8
2.2.2	Retention Policy	8
2.2.3	Controlfile Autobackup	8
2.2.4	Komprimierung einschalten	8
2.2.5	Einstellungen auf Default zurücksetzen	9
2.2.6	Spool	9
2.2.7	Trace File	9
2.2.8	Datum	9
2.3	Recovery Catalog	10
2.3.1	Einrichten	10
2.3.2	Script erstellen	10
2.3.3	Script ändern	12
2.3.4	Script ausführen	12
2.3.5	Script anzeigen	12
2.3.6	Script löschen	13
2.3.7	Wartung	13
2.3.8	Catalog Views	14
2.4	Backup	15
2.4.1	Erstellen	15
2.4.2	Corrupte Blöcke - Backup	16
2.4.3	Backups löschen	16
2.4.4	Database Backup Informationen	17
2.4.5	Archivelog Backup Informationen	18
2.5	Backup Batch	19
2.5.1	Backup Archivelog Batch	19
2.5.2	Backup Database Batch	20
2.6	Flash Recovery Area	22
2.6.1	Überwachung Flash Recovery Area	22
2.6.2	Größe & Auslastung	22
2.7	Restore	23
2.7.1	Restore Backups	23
2.7.2	Restore Controlfile	23
2.8	Kontrolle	25
2.8.1	Kontrolle der Backups	25
2.8.2	Backup Informationen	26
2.8.3	Rman Monitoring	26
2.8.4	Views	28
2.8.5	Stop Rman Job	30
2.8.6	Backup vollständig	30
2.9	Fehlerbehebung	31
2.9.1	Datafile Copy Controlfile Fehlermeldung	31
2.10	Database Klonen lokal	32
2.10.1	Allgemein	32
2.10.2	Backup der Quell Datenbank	32
2.10.3	Erstellen eines Parameter Files	32
2.10.4	Verzeichnisse erstellen	33
2.10.5	Oratab bearbeiten und Oracle_sid setzen	33
2.10.6	Erstellen der Passwort Datei	33
2.10.7	Konfiguration TNS	34
2.10.8	Klonen der Datenbank	34
2.10.9	Erstellen des Server Parameter Files	35
2.10.10	Abschluss	36
2.11	Database Klonen vom Backup	37
2.11.1	Vorarbeiten	37
2.11.2	Erstellen des Pfiles	37
2.11.3	Password Datei	38

2.11.4	Oratab.....	38
2.11.5	Start DB	38
2.11.6	Duplicate DB	38
2.11.7	Check DB.....	39
2.11.8	Tnsnames.ora	39
2.12	Database Klonen Netz.....	40
2.12.1	Allgemein.....	40
2.12.2	Backup der Quell Datenbank	40
2.12.3	Erstellen eines Parameter Files	40
2.12.4	Verzeichnisse erstellen.....	40
2.12.5	Oratab bearbeiten und Oracle_sid setzen	40
2.12.6	Erstellen der Passwort Datei	40
2.12.7	Konfigurartion TNS	40
2.12.8	Klonen der Datenbank	40
2.12.9	Erstellen des Server Parameter Files	41
2.12.10	Abschluss	41
2.13	Disaster Recovery	42
2.13.1	Grundlegendes	42
2.13.2	Erstellen Oracle Service (Windows).....	42
2.13.3	Erstellen der Verzeichnisse.....	42
2.13.4	Restore Spfile.....	43
2.13.5	Restore Control Files	44
2.13.6	Restore Database.....	45
2.13.7	Recover Database.....	46
2.13.8	Restarbeiten.....	46
2.14	Online Backup (Cold Backup)	47
2.14.1	Einzelne Tablespaces	47
2.14.2	Ganze Datenbank	47
3.	Datapump (ab 10G).....	48
3.1	Export	48
3.1.1	Datapump Tabellen / Views.....	48
3.1.2	Vorarbeiten	48
3.1.3	Hilfe anzeigen	48
3.1.4	Export Tabellen.....	48
3.1.5	Export Schemas	49
3.1.6	Export Datenbank	49
3.1.7	Paramter Datei	49
3.1.8	Export Query.....	49
3.1.9	Datapump Jobs löschen.....	50
3.1.10	Parameter	50
3.1.11	Dump Files splitten	50
3.2	Import.....	51
3.2.1	Import Tabellen.....	51
3.2.2	Import Schemas	51
3.2.3	Import Datenbank	51
3.2.4	Parameter Datei.....	51
3.2.5	Remap Data.....	51
3.2.6	Query Abfrage	52
3.2.7	Tabelle vorhanden.....	52
4.	Export / Import (bis 9i).....	53
4.1	Export	53
4.2	Import.....	53
5.	Flashback	54
5.1	Papierkorb	54
5.1.1	Inhalt des Papierkorbs anzeigen.....	54
5.1.2	Inhalt des Papierkorbs löschen.....	54
5.2	Database / Tabelle	54
5.2.1	Table zurücksetzen.....	54
5.2.2	Database zurücksetzen	54
6.	SQL Loader	55
6.1	Aufruf	55

6.2	Beispiele	55
7.	Error & Erro Codes	58
7.1	RMAN-06820	58
8.	Copyright.....	59

1. Vorwort

Diese Dokumentation ist entstanden, da ich beruflich mich mit Oracle beschäftigen musste. Was ich sehr gerne übernommen habe und es macht richtig Spaß mit Oracle zu arbeiten. Alle Informationen, die ich zusammentragen konnte, habe ich nun in dieser Dokumentation geschrieben. Ebenso sind meine Erfahrungen in diese Dokumentation eingeflossen.

Oracle wird auf Linux und Windows Servern in unserer Firma betrieben. Die Installation von Oracle wird für die Linux Server beschrieben, da eine Windows Installation nicht so aufwendig ist.

Diese Dokumentation wurde für die Oracle Datenbank 10G R2 und 11G R1 geschrieben und auch getestet.

Die Datenbank in der Version 11G R1 wurde in einer VMWare Session installiert und als Betriebssystem wurde Novel SLES 10 SP2 installiert.

Bei dem Betriebssystem und auch Oracle handelt es sich um die 32 Bit Version. Für die 64 Bit Version werden noch zusätzliche Softwarepakete gebraucht.

Bei Fragen und Anregungen bin ich unter folgender Mail Adresse zu erreichen:

uwe@seabaer-ag.de



2. Rman

Ein Online Backup der Datenbank kann man nur durchführen, wenn sich die Datenbank im Archivelog Modus befindet. Zum einschalten des Archivelog Modus siehe Dokumentation 03-DB verwalten, Abschnitt 2.7 Archivelog Modus.

Befindet sich die Datenbank nicht im Archive Log Modus, so muss man die Datenbank herunterfahren und im Mount Modus starten. Danach kann man mit Rman die Datenbank sichern und dann die Datenbank öffnen.

```
oracle@woby1002>cat backup.par
shutdown immediate;
startup mount;
backup incremental level=0 as compressed backupset database tag level0 plus
archivelog delete all input;
crosscheck backup;
delete expired backup;
alter database open;
```

2.1 Rman starten

Der Rman wird in der Konsole aufgerufen. Bevor man den Befehl absetzt, muss die ORACLE_SID gesetzt werden und man muss als Oracle User angemeldet sein. Bei dem ersten Aufruf werden die Rman Backups im Controlfile gespeichert und bei dem zweiten Aufruf in einem Catalog, die sich in einer Datenbank befindet.

```
oracle@woby1002:>rman target /
```

```
oracle@woby1002:>rman
rman>connect target /
```

Wird eine Catalog Datenbank verwendet, so wird diese mit mit der Option catalog verbunden.

```
oracle@woby1002:>rman target / catalog <user>/<passwd>@<catdb>
```

```
oracle@woby1002:>rman
rman>connect target / catalog <user>/<passwd>@<catdb>
```

2.2 Konfiguration

2.2.1 Einstellungen Anzeigen

```
rman>show all;

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';
CONFIGURE DEVICE TYPE DISK PARALLELISM 1 BACKUP TYPE TO BACKUPSET; #default
CONFIGURE DATAFILE BACKUP COPIES FOR DEVICE TYPE DISK TO 1; # default
CONFIGURE ARCHIVELOG BACKUP COPIES FOR DEVICE TYPE DISK TO 1; # default
CONFIGURE MAXSETSIZE TO UNLIMITED; # default
CONFIGURE ENCRYPTION FOR DATABASE OFF; # default
CONFIGURE ENCRYPTION ALGORITHM 'AES128'; # default
CONFIGURE COMPRESSION ALGORITHM 'BZIP2'; # default
CONFIGURE ARCHIVELOG DELETION POLICY TO NONE; # default
CONFIGURE SNAPSHOT CONTROLFILE NAME TO '\u02/oradata/scnfcad01.ora';
```

2.2.2 Retention Policy

Mit dieser Konfiguration wird festgelegt, wie lange die Backups verwahrt werden, bevor sie als veraltet (*obsolete*) angesehen werden. Wird der Redundancy Wert z.B. auf 3 erhöht, so wird nach den 4 Backups der erste veraltet.

```
rman>configure retention policy to redundancy 3;
```

Anstelle der Anzahl der zu verwahrenden Backups kann man auch ein Zeitfenster angeben. Das Zeitfenster hat den Vorteil, das man beliebig viel Backups in der Zeit erstellen kann, bevor sie als veraltet angesehen werden. Wird dieser Wert gesetzt, so muss der Wert `control_file_keep_time` in der Datenbank kontrolliert werden. Dieser Wert muss entweder gleich oder größer sein, da sonst die Informationen im Controlfile überschrieben werden. Die Flash Recovery Area sollte auch über einen großen Speicherplatz verfügen.

```
rman>configure retention policy to recovery window of 3 days;
```

2.2.3 Controlfile Autobackup

Mit diesem Parameter kann man einstellen, das der Controlfile bei jedem Backup mitgesichert wird.

```
rman>configure <parameter> <wert>;
rman>configure controlfile autobackup on;
```

2.2.4 Komprimierung einschalten

Dauerhaftes einschalten der Kompression von den Backups und von 2 Chansels.

```
rman>configure device type disk parallelism 2 backup type to compressed
backupset;
```


2.2.5 Einstellungen auf Default zurücksetzen

```
rman>configure <parameter> clear;
```

```
rman>configure <parameter> clean;
```

2.2.6 Spool

Die Syntax für die Ausgabe in einer Datei ist:

```
Spool [log, mslog, trace] [off, to]
```

```
rman>spool log to ListBackup.log  
rman>list backup;  
rman>spool off
```

```
Spooling for log turned off
```

```
Recovery Manager11.1.0.6.0
```

2.2.7 Trace File

Der Trace File wird in dem aktuellen Verzeichnis abgelegt und die Analyse findet mit tkprof dann statt.

```
oracle@woby1002>rman target / log rman.log trace rman.trc debug  
  
rman>sql "alter session set events ``10046 trace name context forever,  
level 12``";  
rman>backup validate datafile 1;  
rman>exit
```

2.2.8 Datum

Standardmäßig wird nur das Datum angezeigt, ohne die Uhrzeit. Möchte man auch die Uhrzeit angezeigt bekommen, so ist in der Shell die Variable NLS_DATE_FORMAT zu setzen.

```
oracle@woby1002>export NLS_DATE_FORMAT="dd.mm.yyyy hh24:mi:ss"
```

2.3 Recovery Catalog

Die Rman Backup Informationen werden in das Controlfile geschrieben und zusätzlich kann man diese Informationen in einem Database Catalog schreiben lassen.

2.3.1 Einrichten

Hierfür sollte man eine eigene Recovery Catalog Instance nehmen, die auch auf einen eigenen Recovery Catalog Server liegen kann. Gibt es einen eigenen Catalog Server, so muss gegebenenfalls die `tnsnames.ora` angepasst werden.

Würde der Catalog in die zu sichernde Datenbank geschrieben, so wäre im Falle einer Beschädigung der Datenbank auch der Catalog verloren.

Anmelden auf der Recovery Catalog Instance.

```
sql>create tablespace ts_rman datafile '<pfad><nam_db>' size 100m
2>autoextend on next 10m maxsize unlimited extent management local
3>autoallocate segment space management auto;

sql>create user rman identified <passwd> temporary tablespace temp
2>default tablespace ts_rman quota unlimited on ts_rman;

sql>grant connect, resource, recovery_catalog_owner to rman;

sql>grant select any dictionary to rman;

oracle@woby1002:>rman target / catalog rman/<passwd>@catdb

rman>create catalog;

rman>register database;

rman>report schema;
```

Existiert eine Standby DB, so wird diese automatisch mit registriert, wenn die Primary DB registriert wird.

Ausgabe, wie die DB-Struktur vor 3 Tagen aussah.

```
rman>report schema at time 'sysdate -3';
```

2.3.2 Script erstellen

Scripte können nur am Rman Prompt erstellt werden. Man muss hierzu mit der Target Datenbank und dem Catalog verbunden sein. Ein Script kann als Local oder Global erstellt werden. Wird ein Globales Script erstellt, so kann dieses Script für alle Datenbanken benutzt werden. Bei einem Local Script nur die Datenbank, mit der man bei der Erstellung verbunden war. Für ein Local oder Global Script kann der gleiche Name genommen werden.

Variablen können in dem Script definiert werden. Ein `&1` definiert die erste Variable, ein `&2` die zweite u.s.w. Special Characters müssen Quotiert werden. Man kann der Variable noch einen Zähler mitgeben, der dann an dem Wert der Variable angehängt wird. Wird zum Beispiel ein `$1.5` definiert und als Wert wird `fullbackup` übergeben, so wird daraus `fullbackup5`. Möchte man als Resultat ein `fullbackup.5` haben, so wird dieses mit einem `&1..5` gemacht. Wird in einem Script Variablen definiert, so muss man Beispiel Werte bei dem erstellen des Scriptes eingeben.

Folgende Syntax Elemente gibt es.

Syntax Element	Beschreibung
GLOBAL	Das Script wird als Global abgespeichert.
SCRIPT <name>	Name des Scriptes. Bei Leerzeichen muss der Name in Hochkommata stehen.
COMMENT '<comment>'	Kommentar für das Script.
backupCommands	Die Befehle werden zwischen { } eingetragen. Es sind alle Befehle erlaubt, ausser @ und @@.
maintenanceCommands	
miscellaneousCommands	
restoreCommands	
FROM FILE '<file_name>'	Liest eine Datei ein. In der ersten Zeile muss ein { und in der letzten Zeile ein } stehen.

Beispiel 2.3.2.1 – Local Script

```
CREATE SCRIPT full_backup
COMMENT "Whole Backup database include archived logs"
{
  BACKUP
  INCREMENTAL LEVEL 0 TAG full_backup
  FORMAT "/disk1/backup/%U
  DATABASE PLUS ARCHIVELOG;
}
RUN { EXECUTE SCRIPT full_backup; }
```

Beispiel 2.3.2.2 – Globales Script

```
CREATE GLOBAL SCRIPT global_full_backup { backup database plus archivelog}
RUN { EXECUTE SCRIPT global_full_backup; }
```

Beispiel 2.3.2.3 – Script mit Variablen

```
CREATE SCRIPT datafile_bck
COMMENT "Backup Datafile with variable"
{
  BACKUP DATAFILE &1 TAG %2.1 FORMAT '/disk1/%3_%U';
}

Enter Value for 1:
Enter Value for 2: datafile_bck
Enter Value for 3:

Starting full resync of recovery catalog
Full resync complete
Created script datafile_bck

RUN { EXECUTE SCRIPT datafile_bck USING 5 dbfile df5; }
```

Das Script wird dann folgendermaßen ausgeführt:

```
BACKUP DATAFILE 5 TAG dbfile1 '/disk1/df5_%U'
```

2.3.3 Script ändern

Die Syntax für das ändern der Scripte ist die gleiche, wie wenn man ein Script erstellt. Anstelle des `create script` wird ein `replace script` benutzt.

```
REPLACE [GLOBAL] SCRIPT <SCRIPT_NAME> [COMMENT] <comment> [{ Commands }]  
[FROM FILE] <file_name>
```

2.3.4 Script ausführen

Ein erstelltes Script wird mit einem `RUN` Befehl und `execute script` ausgeführt. Der `Execute` Befehl kommt nach dem `Run` Befehl und steht zwischen zwei geschweifte Klammern. Die Syntax dafür ist:

```
EXECUTE [GLOBAL] SCRIPT <SCRIPT_NAME> [USING] <PARAMETER>;
```

Die Angabe des Script Names kann auch zwischen zwei ' erfolgen.

Wird bei dem Aufruf `GLOBAL` mit angegeben, so sucht Rman das Script im `Recovery Catalog`. Ist das Script nicht vorhanden, so bekommt man den Fehler `RMAN-06004` zurück. Wird das Script ohne `GLOBAL` aufgerufen, so wird das Script in der `target Database` gesucht. Gibt es hier das Script nicht, so wird im `Recovery Catalog` danach gesucht. Wird es hier gefunden, so wird es dann ausgeführt.

```
rman>RUN {EXECUTE SCRIPT full_backup;}  
rman>RUN {EXECUTE GLOBAL SCRIPT 'df_backup' using 4 Datafile4 df4;}
```

2.3.5 Script anzeigen

Eine Auflistung, welche Scripte im `Recovery Catalog` erstellt worden sind, bekommt man mit `List Script` angezeigt. Wird nur der Befehl `list` abgesetzt, so muss man mit der `Target` und der `Catalog DB` verbunden sein. Mit `list global` werden nur die `Globalen Scripte` und mit `all` werden alle ausgegeben. Bei der Angabe `all` oder `global` muss man nicht mit der `Target DB` verbunden sein, sondern nur mit dem `Catalog`.

```
rman>list script names;  
  
List of Stored Scripts in Recovery Catalog  
  
Global Scripts  
  
Script Name  
Description  
-----  
backup_whole  
whole backup database including archived logs
```

Den Inhalt eines Scriptes kann man sich mit `print script` anzeigen lassen. Die Syntax hierfür ist:

```
PRINT [GLOBAL] SCRIPT <SCRIPT_NAME> [TO FILE] '<FILE_NAME>'
```

```
rman>print script 'backup_whole';

printing stored global script: backup_whole
{
  backup
  incremental level 0 TAG whole_backup
  database plus archivelog
  delete all input;
}
```

2.3.6 Script löschen

Ein erstelltes Script kann man `delete` löschen. Wird `delete script` ohne die Option `global` aufgerufen, so wird nach dem Script in der Target DB gesucht und gelöscht. Ist das Script nicht in der Target DB, so wird nach dem Script in der Catalog DB gesucht. Wird die Option `global` angegeben und das Script ist nicht vorhanden, so bekommt man den Fehler `RMAN-06710` angezeigt.

```
DELETE [GLOBAL] SCRIPT <SCRIPT_NAME>
```

```
rman>delete global script backup_whole;

deleted global script: backup_whole
```

2.3.7 Wartung

Anmelden auf der Recovery Catalog Instance.

```
oracle@woby1002:>sqlplus rman/<passwd>@catdb

sql>select * from cat;
```

Rman Catalog Synchronisieren.

```
rman>resync catalog;
```

Database aus der Registrierung nehmen.

```
rman>unregister database;
```

Catalog auf die neuste Version upgraden.

```
rman>upgrade catalog;
```

Catalog löschen.

```
rman>drop catalog;
```

Die Nachfolgenden Befehle müssen erst noch getestet werden.

```
rman>catalog backuppiece '<filename>';

rman>catalog controlfilecopy '<filename>';
```

```
rman>catalog datafilecopy '<filename>';
rman>catalog archivelog '<filename>';
```

Sucht nach nicht Katalogisierten Backups.

```
rman>catalog recovery area noprompt;
```

Sucht nach Backups im angegebenen Verzeichnis.

```
rman>catalog start with '/u04/oracle/flash_recovery_area/...';
```

2.3.8 Catalog Views

Die Views für den Rman Recovery Catalog beginnen mit rc_. Es gibt mehr als 50 Views und eine Auswahl der Views wird nachfolgend aufgelistet.

```
oracle@tux>sqlplus <connect_catalog_db>
sql>select viewname from user_views where viewname like 'rc_%';
```

View Name	Beschreibung
rc_archived_log	Informationen über archivierte und nicht archivierte redo logs. Korrespondiert mit v\$sarchived_log View.
rc_backup_controlfile	Informationen über die Control Files im Backup Set
rc_backup_corruption	Listet fehlerhafte Blöcke im Datafile Backup auf. Korrespondiert mit v\$backup_corruption View.
rc_database_block_corruption	Informationen über die fehlerhaften Blöcke.
rc_backup_datafile	Informationen über die Datafiles im Backup Set.
rc_backup_piece	Informationen über das Backup.
rc_backup_set	Informationen über das Backup Set.
rc_backup_spfile	Informationen über den Server Parameter File im Backup Set.
rc_controlfile_copy	Informationen über das Controlfile Copy im Backup Set.
rc_database	Zeigt alle registrierten Datenbanken im Catalog an.
rc_database_incarnation	Zeigt Informationen über alle registrierten Datenbanken im Catalog an.
rc_datafile	Informationen über alle registrierten Datafiles.
rc_resync	Informationen über einen Recovery Catalog resync.
rc_rman_configuration	Listet die Rman Konfiguration auf.
rc_rman_status	Alle Informationen über alle Datenbanken Backups.
rc_stored_script	Informationen über gespeicherte Scripte im Catalog.
rc_stored_script_line	

2.4 Backup

2.4.1 Erstellen

- Voll Backup der Datenbank.

Mit `level=0` wird ein full Backup erstellt. `Level 1` erstellt ein Differential Incremental Backup und ein Cumulatives Backup bekommt nach dem `Level 1` noch eine Option mit.

```
rman>backup incremental level=0 as compressed backupset database tag level0;
```

```
rman>backup incremental level=1 as compressed backupset database tag level1;
```

```
rman>backup incremental level=1 cumulative as compressed backupset database tag level1;
```

```
rman>backup database;
```

- Backup eines Tablespaces

```
rman>backup tablespace <tablespace_name>;
```

- Backup einer Datenbank Datei

```
rman>backup datafile <n>;
```

```
rman>backup datafile <datafile_name>;
```

- Backup des Controlfiles

```
rman>backup current controlfile;
```

- Backup des Server Parameter Files

```
rman>backup spfile;
```

- Backup der Archive logs

```
rman>backup archivelog all delete all input;
```

```
rman>backup archivelog from sequence=nnn until sequence= nn [delete {all} input];
```

```
rman>backup as backupset [...];
```

- Backup komprimiert

```
rman>backup as compressed backupset [...];
```

```
rman>backup as copy [...];
```

- Überprüfung der Datenbank auf Fehler

```
rman>backup validate datafile <n>;  
rman>backup validate datafile <datafile_name>;
```

```
rman>backup check logical [...];
```

- Zusätzlich zu den Backup Befehlen können noch Optionen angegeben werden.

```
rman>backup [...] plus archivelog;
```

```
rman>backup [...] include controlfile;
```

```
rman>backup [...] format '/pfad/zum/backup';
```

```
rman>backup [...] tag='beliebiger Text';
```

```
rman>backup [...] not backed up nnn times;
```

```
rman>backup [...] duration hh:mm minimize load;
```

```
rman>backup [...] delete [all] input;
```

2.4.2 Corrupte Blöcke - Backup

Gibt es in der Datenbank Corrupte Blöcke, so bricht Rman das Backup der DB ab. Mit dem set Befehl maxcorrupt gibt man die maximale Anzahl der Corrupten Blöcke pro Datenbank Datei an, bevor Rman das backup abbricht. Diese Set Anweisung muss in einem Run Block stehen.

```
rman>run  
  2>{  
  3> set maxcorrupt for datafile 1 to 10;  
  4> backup incremental level=0 as compressed backupset database tag  
level0;  
  5>}
```

2.4.3 Backups löschen

```
rman>report obsolete;  
rman>delete noprompt obsolete;
```

```
rman>list expired backup;  
rman>delete noprompt expired backup;  
rman>delete force noprompt expired backup;
```

```
rman>delete noprompt expired archivelog all;  
rman>delete force noprompt expired archivelog all;
```

```
rman>change archivelog from sequence = <wert> until sequence = <wert>  
uncatalog;
```


2.4.4 Database Backup Informationen

```
rman>list backup of database;
```

```
rman>list backup of database summary;
```

```
List of Backups
```

```
=====
```

Key	TY	LV	S	DEVICE	TYPE	COMPLETION TIME	#PIECES	#COPIES	COMPRESSED	Tag
160	B	0	A	DISK		25-JAN-12	1	1	YES	LEVEL0
161	B	0	A	DISK		25-JAN-12	1	1	YES	LEVEL0

```
rman>list expired backup;
```

```
rman>list backupset <wert>;
```

```
List of Backup Sets
```

```
=====
```

BS Key	Type	LV	Size	Device	Type	Elapsed Time	Completion Time
1	Incr	0	6.92G	DISK		00:26:41	25.01.2014 03:27:10

PB Key: 1 Status: AVAILABLE Compressed: YES TAG: Level0
 Piece Name: /u04/flash_area/cad10/backupset/o1_mf_nnnd_level0_98.bkp

List of Datafiles in backup set 1

File	LV	Type	Ckp SCN	Ckp Time	Name
5	0	Incr	8686088547	25.01.2014	/u02/oracle/oradata/cad01/example01.dbf
4	0	Incr	8686088547	25.01.2014	/u02/oracle/oradata/cad01/users01.dbf

```
rman>list backupset tag 'level0';
```

Auflisten der Datenbank Instanz

```
rman>report schema;
```

```
Report of datanase schema for database with db_unique_name CAD01
```

```
List of Permanent Datafiles
```

```
=====
```

File	Size(MB)	Tablespace	RB	segs	Datafile Name
1	740	SYSTEM	***		/u02/oracle/oradata/cad01/system01.dbf
2	980	SYSAUX	***		/u02/oracle/oradata/cad01/sysaux01.dbf
3	760	UNDOTBS1	***		/u02/oracle/oradata/cad01/undotbs01.dbf
4	50	USERS	***		/u02/oracle/oradata/cad01/users01.dbf
5	100	EXAMPLE	***		/u02/oracle/oradata/cad01/example01.dbf

```
List of Temporary Files
```

```
=====
```

File	Size(MB)	Tablespace	RB	segs	Datafile Name
1	51	TEMP	***		/u02/oracle/oradata/cad01/temp01.dbf

2.4.5 Archivelog Backup Informationen

```
rman>list backup of archivelog all;
```

```
rman>list backup of archivelog from sequence = <wert> until sequence =  
<wert>;
```

```
rman>list expired archivelog all;
```

```
rman>list expired archivelog from scn = <wert> until scn = <wert>;
```

```
rman>list expired backup of archivelog all;
```

2.5 Backup Batch

2.5.1 Backup Archivelog Batch

Mit dem folgenden Scripten kann man ein Backup der Archivelogs erstellen. Das Script `rman_archivelog.sh` ruft `rman` auf und als Parameterdatei wird die `rman_archivelog.par` verwendet.

Listing rman_archivelog.sh

```
#!/bin/bash
ORACLE_HOME=/u01/app/oracle/product/11.1.0/db_1
ORACLE_SID=cad01
ORACLE_SCRIPTS=/u01/app/oracle/admin/cad01/backup
export ORACLE_HOME ORACLE_SID ORACLE_SCRIPTS
NLS_LANG=AMERICAN_AMERICA.WE8MSWIN1252
NLS_DATE_FORMAT='DD.MM.YYYY HH24:MI:SS'
export NLS_LANG NLS_DATE_FORMAT
DATUM=`date +%Y_%m_%d_%H_%M_%S`
$ORACLE_HOME/bin/rman          CMDFILE=$ORACLE_SCRIPTS/rman_archivelog.par
LOG=$ORACLE_SCRIPTS/log/rman_archivelog_cad01_${DATUM}.log
```

Listing rman_archivelog.par

```
connect target /
backup archivelog all delete input;
delete obsolete;
delete expired backup;
```

Damit die Archivelogs automatisch gesichert werden, kann man mit dem Consolen Befehl `crontab -e` einen automatischen Task erstellen. Soll um 22:00 Uhr jeden Tag ein Backup erfolgen, so trägt man folgende Zeile ein:

```
0 22 * * * /u01/app/oracle/admin/backup/rman_archivelog.sh
```

2.5.2 Backup Database Batch

Ein Full Backup der Datenbank kann mit den nachfolgenden Scripten durchgeführt werden. Mit der Option `tag` im Par-File kann ein beliebiger Text stehen.

Listing rman_level0.sh

```
#!/bin/bash
ORACLE_HOME=/u01/app/oracle/product/11.1.0/db_1
ORACLE_SID=cad01
ORACLE_SCRIPTS=/u01/app/oracle/admin/cad01/backup
export ORACLE_HOME ORACLE_SID ORACLE_SCRIPTS
NLS_LANG=AMERICAN_AMERICA.WE8MSWIN1252
NLS_DATE_FORMAT='DD.MM.YYYY HH24:MI:SS'
export NLS_LANG NLS_DATE_FORMAT
DATUM=`date +%Y_%m_%d_%H_%M_%S`
$ORACLE_HOME/bin/rman          CMDFILE=$ORACLE_SCRIPTS/rman_level0.par
LOG=$ORACLE_SCRIPTS/log/rman_level0_cad01_${DATUM}.log
```

Listing rman_level0.par

```
connect target /
backup incremental level=0 database tag level0 plus archivelog delete all
input;
sql "alter database backup controlfile to trace as
`/u01/app/oracle/admin/cad01/backup/crctl.sql` reuse";
crosscheck backup;
delete obsolete;
delete expired backup;
```

Listing rman_level0A.sh

```
#!/bin/bash
ORACLE_HOME=/u01/app/oracle/product/11.1.0/db_1
ORACLE_SID=cad01
ORACLE_SCRIPTS=/u01/app/oracle/admin/cad01/backup
export ORACLE_HOME ORACLE_SID ORACLE_SCRIPTS
NLS_LANG=AMERICAN_AMERICA.WE8MSWIN1252
NLS_DATE_FORMAT='DD.MM.YYYY HH24:MI:SS'
export NLS_LANG NLS_DATE_FORMAT
DATUM=`date +%Y_%m_%d_%H_%M_%S`
$ORACLE_HOME/bin/rman      target      /      @$ORACLE_SCRIPTS/rman_level0A.par      >
$ORACLE_SCRIPTS/log/rman_level0_cad01_${DATUM}.log
```

Listing rman_level0A.par

```
run {
  configure controlfile autobackup on;
  sql 'alter system switch logfile';
  backup incremental level=0 database tag level0 plus archivelog delete all
input;
  sql "alter database backup controlfile to trace as
`/u01/app/oracle/admin/cad01/backup/crctl.sql` reuse";
  crosscheck backup;
  delete obsolete;
  delete expired backup;
}
```

Parameter File für ein Cold Backup.

```
run {
  shutdown immediate
  startup mount
  configure controlfile autoback on;
  configure device type disk parallelism 4 backup type to compressed
  baakupset;
  configure snapshot controlfile name to '/u01/backup/snapcf_cad10.f';
  show all;
  backup incremental level=0 database tag level0 plus archivelog delete all
  input;
  crosscheck backup;
  delete obsolete;
  delete expired backup;
  sql "alter database open";
}
```

Mehrere Channels kann man folgendermaßen definieren.

```
run {
  allocate channel ch1 device type disk;
  allocate channel ch2 devise type disk;
  allocate channel ch3 device type disk;
  allocate channel ch4 device type disk;
  backup incremental level=0 database tag level0 plus archivelog delete all
  input;
  release channel ch1;
  release channel ch2;
  release channel ch3;
  release channel ch4;
}
```

2.6 Flash Recovery Area

2.6.1 Überwachung Flash Recovery Area

Einen Überblick über den Status der Flash Recovery Area erhält man mit folgenden Abfragen.

```
sql>select * from v$flash_recovery_area_usage;
```

FILE_TYPE	PERCENT_SPACE_USED	PERCENT_SPACE_RECLAIMABLE	NUMBER_OF_FILES
CONTROL_FILE	0	0	0
REDO LOG	0	0	0
ARCHIVED LOG	,71	0	27
BACKUP PIECE	10,93	,09	7
IMAGE COPY	0	0	0
FLASHBACK LOG	,54	0	106
FOREIGN ARCHIVED LOG	0	0	0

7 Zeilen ausgewählt

```
sql>select sum(percent_space_used) "Total %" from
2>v$flash_recovery_area_usage;
```

Total %
12,68

2.6.2 Größe & Auslastung

Die Größe und die Auslastung der Flash Recovery Area kann man sich mit der folgenden Abfrage anzeigen lassen.

```
sql>col name for a40
sql>col space_limit for a11
sql>col space_used for a10

sql>select name,
2>(space_limit/1024/1024)|| 'MB' as "Space_Limit",
2>round(space_used/1024/1024) || 'MB' as "Space_Used"
2>from v$recovery_file_dest;
```

NAME	Space_Limit	Space_Used
/u01/app/oracle/fast_recovery_area	4182MB	888MB

2.7 Restore

2.7.1 Restore Backups

Gibt man nach dem Befehl `restore` die Option `validate` mit an, so wird nur ein Test-Restore durchgeführt.

```
rman>restore validate database;
rman>restore valiadte tablespace users;
rman>restore validate controlfile;
rman>restore validate archivelog all;
rman>restore validate archivelog sequence = 1600;
rman>restore validate archivelog from time 'sysdate -1';
rman>restore valiadate spfile;
```

Ohne die Option `validate` wird ein Restore durchgeführt.

```
rman>restore archivelog from sequence = <wert> until sequence <wert>;
```

2.7.2 Restore Controlfile

Einen Restore von den Controlfile kann mit oder ohne einen Catalog gemacht werden. In beiden Fällen muss sich die Datenbank im `nomount` Status befinden.

Ohne Catalog Datenbank.

```
oracle@woby1002>rman target /
rman>startup force nomount;
ORACLE instance started
Total System Global Area 3223613440 bytes
Fixed Size                2148680 bytes
Variable Size             2013267640 bytes
Database Buffers         1191182336 bytes
Redo Buffers              17014784 bytes
rman>restore controlfile from
  2>' /u04/app/oracle/flash_recovery_area/cad10/autobackup/2012_11_17/
  3> o1_mf_s_799558311_8bfxt0k.bkp' ;
Starting restore at 09-JAN-13
allocated channel ORA_DISK_1
channel ORA_DISK_1: SID=1087 device type=DISK
```

Mit Catalog Datenbank

```
oracle@woby1002>rman target / catalog rman/rman@catdb
rman>startup force nomount;

ORACLE instance started

Total System Global Area 3223613440 bytes

Fixed Size                2148680 bytes
Variable Size             2013267640 bytes
Database Buffers         1191182336 bytes
Redo Buffers              17014784 bytes

rman>restore controlfile;

Starting restore at 09-JAN-13
allocated channel ORA_DISK_1
channel ORA_DISK_1: SID=1087 device type=DISK
```


2.8 Kontrolle

2.8.1 Kontrolle der Backups

```
rman>crosscheck backup;
```

```
rman>crosscheck archivelog all;
```

```
rman>crosscheck backup of archivelog all;
```

Um Backups zu überprüfen, gibt es den Befehl `validate` im Rman.

```
rman>validate archivelog all;

rman>validate archivelog sequence 5;

rman>validate controlfilecopy
  2>' /u01/app/oracle/admin/cad01/backup/crctl.sql';

rman>validate tablespace 'users01';
```

```
rman>validate current controlfile;

Starting validate at 16-JUN-12
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=944 device type=DISK
channel ORA_DISK_1: starting validation of datafile
channel ORA_DISK_1: specifying datafile(s) for validation
including current control file for validation
channel ORA_DISK_1: validation complete, elapsed time: 00:00:02
List of Control File and SPFILE
=====
File Type      Status Blocks Failing Blocks Examined
-----
Control File OK      0          988
Finished validate at 16-JUN-12
```

Informationen über das Backup kann man sich mit dem Befehl `list` anzeigen lassen.

```
rman>list backup summary;

using target database control file instead of recovery catalog

List of Backups
=====
Key    TY LV S Device Type Completion Time #Pieces #Copies Compressed Tag
-----
1426   B  0  A DISK          13-Jul-14      1      1      YES  LEVEL0
1427   B  0  A DISK          13-Jul-14      1      1      YES  LEVEL0
```

```
rman>list expired backup summary;

rman>report schema;

Report of database schema for database with db_unique_name CAD10

List of Permanent Datafiles
=====
File Size(MB) Tablespace RB segs Datafile Name
-----
1      830      SYSTEM      ***      /u01/app/oracle/oradata/cad10/system01.dbf
2     11300     SYSAUX      ***      /u01/app/oracle/oradata/cad10/sysaux01.dbf
3      845     UNDOTBS1    ***      /u01/app/oracle/oradata/cad10/undo01.dbf
4       36      USERS      ***      /u01/app/oracle/oradata/cad10/users01.dbf

List of Temporary Files
=====
File Size(MB) Tablespace RB segs Datafile Name
-----
1      306      TEMP      ***      /u01/app/oracle/oradata/cad10/temp01.dbf
```

2.8.2 Backup Informationen

```
sql>alter session set nls_date_format='DD.MM.YYYY hh24:mi:ss';
sql>col session_recid for 999999 heading "SESSION|RECID"
sql>col session_stamp for 99999999999 heading "SESSION|STAMP"
sql col input_type for a10 heading "INPUT|TYPE"

sql>select session_recid, session_stamp, start_time, end_time,
2>status, input_type from v$rman_backup_job_details
3>where start_time > trunc(sysdate)-1 order by start_time;
```

SESSION RECID	SESSION STAMP	START_TIME	END_TIME	INPUT STATUS	TYPE
16796	8046252	08.12.2012 08:00:00	08.12.2012 08:15:00	FAILED	SPFILE

2.8.3 Rman Monitoring

Wenn ein Rman Job läuft, kann man den Job mit dem nachfolgenden Statement überwachen. Am besten ist es, diese Abfrage in einer Sql-Datei zu schreiben und dann mehrmals aufzurufen.

```
sql>set lines 200
sql>select sid, serial#, context, sofar, totalwork,
2>round(sofar/totalwork*100,2) "% Complete"
3>from v$session_longops
4>where opname like 'RMAN%'
5>and opname not like '%aggregate%'
6>and totalwork != 0
7>and sofar <> totalwork;
```

SID	SERIAL#	CONTEXT	SOFAR	TOTALWORK	% Complete
4	23	1	12765	43265	29.50

```

sql>col operation for a10
sql>select operation, start_time, end_time, object_type, status
  2>from v$rman_status where status = 'RUNNING';

```

OPERATION	START_TIME	END_TIME	OBJECT_TYPE	STATUS
RMAN	03.11.2012 08:43:12	03.11.2012 08:56:23		RUNNING
Backup	03.11.2012 08:50:44	03.11.2012 08:56:23	DB INCR	RUNNING

```

sql>col sid for 9999
sql>col spid for 99999
sql>col client_info for a25
sql>col event for a90
col>col secs for 9999

sql>select sid, spid, client_info, event, seconds_in_wait secs, p1, p2, p3
  2>from v$process p, v$session s where p.addr = s.addr
  3>and client_info like 'rman channel%';

```

SID	SPID	CLIENT_INFO	EVENT	SECS	P1	P2	P3
909	7824	rman channel=ORA_DISK_2	SQL*Net message	2496	873465	1	0
1012	7820	rman channel=ORA_DISK_1	db file sequent	2491	1	6456	1

Auflistung des letzten Backups (Log Informationen).

```

sql>select output from gv$rman_output
  2>where session_recid = (select max(session_recid)
  3>from gv$rman_output);

```

Oder in einer anderen Abfrage.

```

sql>select gv.output from gv$rman_output gh,
  2>(select max(session_recid) as maxid
  3>from gv$rman_output) maxrecid
  4>where gv.session_recid = maxrecid.maxid;

```

2.8.4 Views

Informationen über die Rman Backup Jobs stehen in folgenden Views.

- v\$archived_log Archived and unarchived redo logs
- v\$backup_archivelog_details Details about archived redo log backup
- v\$backup_archivelog_summary Summary of Information about archived redo log
- v\$backup_controlfile Control files backed up in backup set
- v\$backup_controlfile_details Details about control file backups
- v\$backup_controlfile_summary Summary of information about control file
- v\$backup_copy_details Details about datafile image copy
- v\$backup_copy_summary Summary of information about datafile image copy
- v\$backup_corruption Corrupt block ranges in datafile backups
- v\$backup_datafile Datafiles in backup set.
- v\$backup_datafile_details Detail about datafile backup
- v\$backup_datafile_summary Summarx of information about datafile backups
- v\$backup_files Rman backups and copies known to the repository
- v\$backup_piece Backup pieces
- v\$backup_piece_details Details about backup pieces
- v\$backup_redolog Archived redo logs in backup sets
- v\$backup_set Backup sets for all incarnation of db's registered in the catalog
- v\$backup_set_details Details about backup sets
- v\$backup_set_summary Summary information about backup sets
- v\$backup_sfile Server parameter files in backups
- v\$backup_sfile_details Details about SPFILE backups
- v\$backup_sfile_summary Summary of information about SPFILE backups
- v\$controlfile_copy Control file copies on disk
- v\$copy_corruption Corrupt block ranges in datafile copies
- v\$database Databases registered in the recovery catalog
- v\$database_block_corruption Database blocks marked as corrupted in backup or copy
- v\$database_incarnation Database incarnations registered in the recovery catalog
- v\$datafile Datafiles registered in the recovery catalog
- v\$datafile_copy Datafile copies on disk
- v\$log_history Online redo log history indicatin when log switches occurred
- v\$offline_range Offline ranges for datafiles
- v\$rman_output Output from rman commands
- v\$proxy_archivelog Archived log backups taken with ptoxy copy functionality
- v\$proxy_archivelog_details Details about proxy archived redo logs
- v\$proxy_archivelog_summary Summary of information about proxy archived redo logs
- v\$proxy_controlfile Control file backups taken with proxy copy functionality
- v\$proxy_copy_details Details about datafile proxy copies
- v\$proxy_copy_summary Summary information about datafile proxy copies
- v\$proxy_datafile Datafile backups where taken using proxy copy functionality
- v\$log und v\$logfile Online redo logs for all incarnation of the database
- v\$thread All redo threads for all incarnations of the database
- v\$rman_backup_job_details Details about backup jobs
- v\$rman_backup_subjobs_details Details about backup subjobs
- v\$backup_type Used internally by Enterprise Manager
- v\$rman_configuration Rman configuration settings
- v\$rman_status Historical status information about Rman operations
- v\$unusable_backupfile_details Unusable backup files registered in the recovery catalog
- gv\$rman_output

Beispiele:

```

sql>col name for a30
sql>col value for a70

sql>select * from v$rman_configuration;

```

CONF#	NAME	VAULE
1	CONTROLFILE AUTOBACKUP	ON
2	DEVISE TYPE	DISK PARRALLESIM 2 BACKUP TYPE TO COMP..
3	SNAPSHOT CONTROLFILE NAME TO	'/u01/app/oracle/snap/snapcf_cad10.f'

```

sql>select
  2> recid,
  3> parent_recid,
  4> command_id,
  5> object_type,
  6> operation,
  7> status,
  8> start_time,
  9> end_time
10>from
11> v$rman_status;

```

RECID	PARENT_RECID	COMMAND_ID	OBJECT_TYPE	OPERATION	STATUS	STA
10140	10138	2017-04-29T21:05:04	DB INCR	BACKUP	COMPLETED	201

```

sql>col backup_type for a12
sql>col file_type for a13
sql>col fname for a102
sql>col device_type for a10 heading 'Device|Type'

```

```

sql>select
  2> pkey,
  3> bs_completion_time,
  4> backup_type,
  5> file_type,
  6> status,
  7> fname,
  8> device_type
  9>from
10> v$backup_files
11>order by
12> device_type;

```

PKEY	BS_COMPLETION_TIM	BACKUP_TYPE	FILE_TYPE	STATUS	FNAME
5		COPY	CONTROLFILE	AVAILABLE	/u01/app/oracle
11651	08.05.17 21:08:11	BACKUP SET	PIECE	AVAILABLE	/u01/fast_rec...

2.8.5 Stop Rman Job

Die laufenden Rman Jobs kann man mit dem nachfolgenden Sql-Statement sich anzeigen lassen und anschließend mit `alter system kill session` löschen.

```
sql>alter system set nls_date_format='DD.MM.YYYY hh24:mi:ss';
sql>col spid for a8
sql>col client_info for a40 heading 'Client|Info'

sql>select a.spid, b.sid, b.serial#, b.logon_time,
2>c.seconds_in_wait "Wait", c.state, client_info
3>from v$session_wait c, v$session b, v$process a
4>where b.client_info like 'rman%'
5>and b.sid = c=sid
6>and b.paddr = a.addr
7>order by b.logon_time;

          CLIENT
SPID      SID SERIAL# LOGON_TIME          WAIT  STATE  INFO
-----
3972      958      326 11.02.2013 08:10:12 219365 WAITING rman channel1=ORA

sql>alter system kill session '958,326';

System wurde geändert
```

2.8.6 Backup vollständig

```
rman>report unrecoverable;

using target database control file instead of recovery catalog
Report of files that need backup due to unrecoverable operations
File Type of Backup Required Name
-----

rman>report need backup;

RMAN retention policy will be applied to the command
RMAN retention policy ist set to redundancy 1
Report of files with less than 1 redundant backups
File #bkps Name
-----
```

2.9 Fehlerbehebung

2.9.1 Datafile Copy Controlfile Fehlermeldung

```
rman>crosscheck controlfilecopy
'/u01/app/oracle/product/11.1.0/db_1/dbs/snapcf_cad01.f';
```

Falls die Datei nicht gefunden wird, so muss der nachfolgende Befehl abgesetzt werden.

```
rman>crosscheck datafilecopy
'/u01/app/oracle/product/11.1.0/db_1/dbs/snapcf_cad01.f';

rman>report obsolete;
```

Wird die Datei nach dem vorherigen Befehl angezeigt, so muss die Datei aus dem Catalog gelöscht werden.

```
rman>change controlfilecopy
'/u01/app/oracle/product/11.1.0/db_1/dbs/snapcf_cad01.f' uncatalog;
```

Achtung:

Die Datei darf **nicht** von der Festplatte gelöscht werden.

2.10 Database Klonen lokal

2.10.1 Allgemein

Um eine Datenbank mit Rman zu klonen, sind insgesamt 9 Schritte nötig.

- Backup der Quell Datenbank
- Erstellen eines Parameter Files
- Verzeichnisse erstellen
- Oratab bearbeiten und Oracle_sid setzen
- Erstellen einer Passwort Datei
- Konfiguration TNS
- Klonen der Datenbank
- Erstellen des Server Parameter Files
- Abschluss

2.10.2 Backup der Quell Datenbank

Als erstes wird ein Backup der zu klonenden Datenbank gebraucht. Ist ein Backup vorhanden, so kann dieses Backup verwendet werden. Nachfolgend ein Beispiel für ein Backup.

```
oracle@woby1002>rman target sys@<oracle_sid> nocatalog
rman>backup database plus archivelog format '/u01/ora_backup/%d_%u_%s';
```

Alternativ kann man auch ohne ein Backup die Datenbank klonen, siehe 2.9.8

2.10.3 Erstellen eines Parameter Files

Nun erstellen wir aus der Quell Datenbank ein Pfile und fügen anschließend zwei Zeilen ein. Gibt es in der Quell Datenbank verschiedene Pfade zu den Datenbank Dateien, so sind diese alle in den convert Parameter aufzunehmen. Alle vorhandennnen Pfadangaben mit der alten Oracle_sid werden auf die neue abgeändert. Die Pfadangaben erfolgen jeweils mit einem abschließenden Backslash.

```
sql>create
  2>pfile='/u01/app/oracle/product/11.2.0/db_1/dbs/init<new_sid>.ora'
  3>from spfile;

File created

sql>exit

oracle@woby1002>vi init<new_sid>.ora

db_file_name_convert=('<quell_db_path>', '<ziel_db_path>', '<quell_db_path>',
 '<ziel_db_path>')
```



```
log_file_name_convert=('<quell_db_path>','<ziel_db_path>','<quell_db_path>','<ziel_db_path>')
```

Wird eine Sqlplus Session aus dem Verzeichnis \$ORACLE_HOME/dbs aufgerufen, so kann auch die verkürzte Version genommen werden.

```
sql>create pfile='init<new_sid>.ora' from spfile;
```

2.10.4 Verzeichnisse erstellen

Alle Verzeichnisse aus der Quell Datenbank müssen angelegt werden, da rman sie nicht erstellt. Im Verzeichnis \$ORACLE_BASE/admin werden die Audit Verzeichnisse erstellt. Ebenso die Verzeichnisse für die Tablespace Dateien und den Redo Log Dateien.

```
oracle@woby1002>cd $ORACLE_BASE/admin
oracle@woby1002:/u01/app/oracle/admin>mkdir <new_sid>
oracle@woby1002:/u01/app/oracle/admin>cd <new_sid>
oracle@woby1002:/u01/app/oracle/admin/<new_sid>>mkdir
{adump,bdump,arch,backup,cdump,dpdump,pfile,udump}
oracle@woby1002>mkdir /u02/oracle/oradata/<new_sid>
oracle@woby1002>mkdir /u03/oracle/oradata/<new_sid>
```

2.10.5 Oratab bearbeiten und Oracle_sid setzen

Im Verzeichnis /etc befindet sich die Datei oratab. In dieser Datei wird nun die neue Datenbank hinzugefügt. Anschließend wird die Variable Oracle_sid mit dem neuen Wert gesetzt.

```
oracle@woby1002>vi /etc/oratab
<new_sid>:/u01/app/oracle/product/11.2.0/db_1:y
oracle@woby1002>export ORACLE_SID=<new_sid>
```

2.10.6 Erstellen der Passwort Datei

Eine Passwort Datei wird mit dem Befehl orapwd erstellt. Diese Datei muss im Verzeichnis \$ORACLE_HOME/dbs liegen.

```
oracle@woby1002>orapwd file=${ORACLE_HOME}/dbs/orapw${ORACLE_SID}
password=<sys_pwd>
```

2.10.7 Konfiguration TNS

Die neue Datenbank wird in der `tnsnames.ora`, die sich im Verzeichnis `$ORACLE_HOME/network/admin` befindet, hinzugefügt.

```
oracle@woby1002>cat tnsnames.ora

<new_sid> =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL=TCP)(HOST=<name>)(PORT=1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = <new_sid>)
    )
  )
```

In der `listener.ora` sollten beide Datenbanken in der `SID_LIST_LISTENER` eingetragen sein.

```
oracle@woby1002>cat listener.ora

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = <oracle_sid>)
      (ORACLE_HOME = /u01/app/oracle/product/11.2.0/db_1)
      (SID_NAME = <oracle_sid>)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = <new_sid>)
      (ORACLE_HOME = /u01/app/oracle/product/11.2.0/db_1)
      (SID_NAME = <new_sid>)
    )
  )
```

Nun wird der Listener einmal neu gestartet.

```
oracle@woby1002>lsnrctl stop
oracle@woby1002>lsnrctl start
```

2.10.8 Klonen der Datenbank

Nun können wir die Datenbank klonen. Als erstes starten wir die Datenbank mit der `nomount` Option.

```
oracle@woby1002>sqlplus / as sysdba
sql>startup nomount
```

Nach dem Starten der Datenbank rufen wir `rman` auf und klonen die Datenbank

```
oracle@woby1002>rman target sys@<quell_db> nocatalog auxiliary /
rman>duplicate target database to <new_sid>;
```

Das klonen der Datenbank kann auch ohne ein vorheriges Backup der Quell Datenbank gemacht werden. Hierbei ist zu beachten, dass die Passwörter für `sys` auf beiden Datenbanken gleich sein müssen.

```
oracle@woby1002>rman      target      sys@<quell_db>      nocatalog      auxiliary
sys@<ziel_db>

rman>duplicate target database to <new_sid> from active database;
```

Möchte man einen Tablespace nicht mit klonen, so gibt man ein `skip tablespace` mit an.

```
rman>run
rman>{
rman>duplicate target database to <new_sid> from active database
rman>skip tablespace test;
rman>}
```

Wird das klonen der Datenbank mit der nachfolgenden Fehlermeldung abgebrochen, so ist die Variable `file_name_convert` in der `init<new_sid>.ora` zu überprüfen.

```
rman-05001: auxiliary filename '%s' conflicts with a file used by the
target database
```

War das klonen der Datenbank erfolgreich, so wird folgende Meldung angezeigt.

```
Database opened
Finished Duplicate DB at 26-Jun-11
```

2.10.9 Erstellen des Server Parameter Files

Zum Abschluss erstellen wir noch den Server Parameter File und starten die Datenbank einmal neu. Damit der Verkürzte Aufruf auch funktioniert, sollte man `Sqlplus` aus dem Verzeichnis `$ORACLE_HOME/dbs` aufrufen.

```
oracle@woby1002>sqlplus / as sysdba

sql>create spfile from pfile;

File created

sql>startup force
```

Die nicht mehr gebrauchten Einträge für `file_name_convert` werden nun aus dem System gelöscht.

```
sql>alter system reset db_file_name_convert scope=spfile sid='*';

sql>alter system reset log_file_name_convert scope=spfile sid='*';
```

2.10.10 Abschluss

Da sich die Quell Datenbank im `archive log` Modus befunden hat, so ist auch die neue im `archive log` Modus. Daher ist es ratsam, ein Backup für die neue Datenbank anzulegen. Ist ein Backup nicht erwünscht, so schaltet man den `archive log` Modus aus.

```
sql>shutdown immediate
sql>startup mount
sql>alter database noarchivelog;
sql>alter database open;
```

2.11 Database Klonen vom Backup

Eine neue Instanz soll erstellt werden, aber nicht von einer aktuellen Instanz, sondern von einem alten Stand der Datenbank. Dieses kann auch auf der Maschine erfolgen, auf der die Instanz läuft.

2.11.1 Vorarbeiten

Als erstes wird ein Rman Backup gebraucht. Dieses kann ein aktuelles Backup sein oder ein älteres Backup der Datenbank. Das Backup muss incl. der Controlfiles gemacht worden sein. Dieses Backup wird in die `fast_recovery_area` für die neue Instanz kopiert.

```
oracle@woby1002>mkdir /u01/app/oracle/fast_recovery_area/cad99

oracle@woby1002>cd /u01/app/oracle/fast_recovery_area/cad99

oracle@woby1002>cp -r /u01/app/oracle/fast_recovery_area/cad10/archivelog
oracle@woby1002>cp -r /u01/app/oracle/fast_recovery_area/cad10/autobackup
oracle@woby1002>cp -r /u01/app/oracle/fast_recovery_area/cad10/backupset
oracle@woby1002>cp -r /u01/app/oracle/fast_recovery_area/cad10/flashback
oracle@woby1002>cp -r /u01/app/oracle/fast_recovery_area/cad10/onlinelog
```

2.11.2 Erstellen des Pfiles

Nun erstellen wir einen Pfile und kopieren ihn.

```
sql>create pfile from spfile;

File created.

sql>exit

oracle@woby1002>cp $ORACLE_HOME/dbs/initcad10.ora
$ORACLE_HOME/dbs/initcad99.ora
```

In den Pfile für die neue Instanz ersetzen wir die alte `ORACLE_SID` durch die neue.

```
oracle@woby1002>grep cad99 initcad99.ora

*.dispatcher = `(PROTOCOL=TCP) (SERVICE=cad99XDB)`
db_name = `cad99`
audit_file_dest = `/u01/app/oracle/admin/cad99/adump`
control_files =
`/u01/app/oracle/oradata/cad99/control01.ctl`,`/u01/app/oracle/fast_recover
y_area/cad99/control02.ctl`
```

Ausserdem fügen wir noch zwei Parameter dem Pfile hinzu.

```
oracle@woby1002>grep convert initcad99.ora

db_file_name_convert =
(`/u01/app/oracle/oradata/cad10`,`/u01/app/oracle/oradata/cad99`)
log_file_name_convert =
(`/u01/app/oracle/oradata/cad10`,`/u01/app/oracle/oradata/cad99`)
```

2.11.3 Password Datei

Die Password Datei von der zu klonenen DB kopieren wir für die neue DB.

```
oracle@woby1002>cd $ORACLE_HOME/dbs
oracle@woby1002>cp orapwcad10 orapwcad99
```

2.11.4 Oratab

Die neue DB wird der oratab in /etc hinzugefügt.

```
oracle@woby1002>grep cad99 /etc/oratab
cad99:/u01/app/oracle/product/11.2.0/dbhome_1:y
```

Unter Windows muss ein Dienst angelegt werden.

```
C:\Oracle>oradim -new -sid cad10 [-intpwd <passwd>] -startmode auto -
spfileInstance created
```

2.11.5 Start DB

Nun bringen wir die neu DB in den nomount status und erstellen einen Spfile.

```
oracle@woby1002>export ORACLE_SID=cad99
oracle@woby1002>sqlplus / as sysdba
sql>startup nomount
sql>create spfile from pfile;
```

2.11.6 Duplicate DB

Jetzt können wir die neu DB aus dem rman Backup erstellen. Der erste duplicate Befehl wird genommen, wenn die Datenbank File Struktur eine andere ist als die Source DB.

```
oracle@woby1002>rman auxiliary /
rman>duplicate target database to cad99 backup location
'/u01/app/oracle/fast_recovery_area/';
rman>duplicate target database to cad99 backup location
'/u01/app/oracle/fast_recovery_area/' nofilenamecheck;
```

Gibt es in der Fast Recovery Area mehrere Instanzen, so wird für den Restore das erste aus der Verzeichnis Liste genommen. Soll aber ein anderes Backup für den Restore genommen werden, so sind die Dateien in ein anderes Verzeichnis zu kopieren z.B. /u01/app/oracle/restore.

2.11.7 Check DB

Die neue DB hat natürlich auch eine neue DBID bekommen.

```
oracle@woby1002>sqlplus / as sysdba  
sql>select dbid from v$database;
```

2.11.8 Tnsnames.ora

Zum Schluss fügen wir die neue Instanz der `tnsnames.ora` hinzu.

2.12 Database Klonen Netz

2.12.1 Allgemein

Das Klonen einer Datenbank über das Netz erfolgt genauso wie unter 2.9 beschrieben. Es gibt nur zwei Ergänzungen, die für das klonen über Netz gemacht werden müssen. Alle Arbeiten werden auf dem neuen System durchgeführt.

2.12.2 Backup der Quell Datenbank

Beschreibung siehe 2.9.2.

2.12.3 Erstellen eines Parameter Files

Die Erstellung des Parameter Files geschieht auf der Target Datenbank und der File wird anschließend auf den Ziel Server kopiert.

Beschreibung siehe 2.9.3.

2.12.4 Verzeichnisse erstellen

Beschreibung siehe 2.9.4.

2.12.5 Oratab bearbeiten und Oracle_sid setzen

Beschreibung siehe 2.9.5

2.12.6 Erstellen der Passwort Datei

Beschreibung siehe 2.9.6

2.12.7 Konfigurartion TNS

Beschreibung siehe 2.9.7.

Hier wird zusätzlich die Quell Datenbank eingetragen.

2.12.8 Klonen der Datenbank

Damit das klonen funktioniert, muss das Backup von der Quell Datenbank auf dem anderen Server / Rechner kopiert werden.

Wird das laufende System dupliziert, so braucht man ds Backup nicht kopiert zu werden.

Die Pfade müssen auf dem Ziel genauso sein, wie auf dem Quell Server.

Beschreibung siehe 2.9.8.

2.12.9 Erstellen des Server Parameter Files

Beschreibung siehe 2.9.9.

2.12.10 Abschluss

Beschreibung siehe 2.9.10.

2.13 Disaster Recovery

In diesem Beispiel wird angenommen, das der Server vollständig ausgefallen ist und ein Neuaufbau auf einer anderen Hardware vorgenommen wird.

2.13.1 Grundlegendes

Damit der Restore / Neuaufbau der Datenbank funktioniert, muss man folgende Backups der Datenbank haben:

- Ein Rman Full Backup
- Die Rman ArchiveLog Dateien
- Controlfile und Spfile autobackup

Folgende Schritte für einen Restore sind nötig:

- Erstellen des Oracle Services (Windows)
- Erstellen aller benötigten Verzeichnisse
- Restore des Spfiles aus dem autobackup
- Restore Controlfile aus dem autobackup
- Restore Database aus dem Rman Backup
- Recover Database

2.13.2 Erstellen Oracle Service (Windows)

Auf einer Windows Plattform muss als erstes der Oracle Service erstellt werden. Die Password Datei muss unter Windows den Namen `pwd<ORACLE_SID>.ora` haben. Wird die Password Datei vorher erstellt, so braucht bei dem anlegen des Services die Option `-intpwd` nicht mit angegeben werden.

```
C:\Oracle>orapwd -file=pwdcad10.ora password=Oracle  
  
C:\Oracle>oradim -new -sid cad10 [-intpwd <passwd>] -startmode auto -spfile  
Instance created
```

2.13.3 Erstellen der Verzeichnisse

Damit der Restore erfolgreich durchgeführt werden kann, müssen die Verzeichnisse vorher angelegt sein. In dem Flash Recovery Area Verzeichnis wird nach dem erstellen die Dateien aus der Sicherung abgelegt.

```
oracle@woby1002>cd $ORACLE_BASE/admin/<oracle_sid>  
oracle@woby1002>mkdir {bdump,cdump,udump}  
  
oracle@woby1002>cd /u02/oradata  
oracle@woby1002>mkdir cad10  
  
oracle@woby1002>mkdir -p /u04/app/oracle/flash_recovery_area/cad10
```

2.13.4 Restore Spfile

Für das Restore des Spfiles wird die Database ID (DBID) gebraucht. Wurde das Autobackup Format CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TY DISK TO '%F' gesetzt, so hat das Autobackup File das Format c-#####-YYMMDD-QQ. Der Platzhalter ##### steht für die DBID. Die Pfadangabe zur Lage des Spfiles kann entfallen, wenn der Ablageort der Standardmäßige sein soll.

```

oracle@woby1002>export ORACLE_SID=cad10
oracle@woby1002>rman target /

Recovery Manager: Release 11.1.0.6.0 - Production on Wed Jan 9 11:50:20
2013

Copyright © 1982, 2007, Oracle. All rights reserved.

Connect to target database (not started)

rman>set dbid=563086973;

executing command: SET DBID

rman>startup force nomount;

startup failed: ORA-01078: failure in processing system parameters
LRM-00109: could not open parameter file
'/u01/app/oracle/product/11.1.0/db_1/dbs/initcad10.ora'

Starting Oracle instance without parameter file for retrieval of spfile
Oracle instance started

Total System Global Area 158662656 bytes

Fixed Size          2142496 bytes
Variable Size       83889888 bytes
Database Buffers    67108864 bytes
Redo Buffers        5521408 bytes

rman>restore spfile to
 2>' /u01/app/oracle/product/11.1.0/db_1/dbs/spfilecad10.ora' from
 3>' /u04/app/oracle/flash_recovery_area/cad10/autobackup/2012_11_17/
 4>o1_mf_s_799558311_8bfxt0k.bkp';

starting restore at 09-JAN-2013
using chanel ORA_DISK_1

channel          ORA_DISK_1:                autobackup                found:
/u04/app/oracle/flash_recovery_area/cad10/autobackup/2012_11_17/
o1_mf_s_799558311_8bfxt0k.bkp
channel ORA_DISK_1: SPFILE restore from autobackup complete
Finished restore at 09-JAN-13

rman>shutdown immediate

```

Ohne die Angabe der DBID, kann man mit DB_NAME den Spfile wieder herstellen.

```

rman>restore spfile to
 2>' /u01/app/oracle/product/11.1.0/db_1/dbs/spfilecad10.ora' from
 3>' /u04/app/oracle/flash_recovery_area/cad10/autobackup/2012_11_17/
 4>o1_mf_s_799558311_8bfxt0k.bkp' DB_NAME 'CAD10';

```

Einen Restore kann man auch mit dem Parameter `db_recovery_file_dest` durchführen.

```
rman>restore spfile from autobackup
 2>db_recovery_file_dest='/u01/app/oracle/fast_recovery_area'
 3>db_name='cad10';
```

2.13.5 Restore Control Files

Nachdem der `Spfile` aus dem Autobackup zurückgespielt worden ist, kann man die Datenbank im `nomount` Status hochfahren.

```
rman>startup force nomount;

ORACLE instance started

Total System Global Area 3223613440 bytes

Fixed Size                2148680 bytes
Variable Size             2013267640 bytes
Database Buffers         1191182336 bytes
Redo Buffers              17014784 bytes

rman>restore controlfile from
 2>' /u04/app/oracle/flash_recovery_area/cad10/autobackup/2012_11_17/
 3> o1_mf_s_799558311_8bfxt0k.bkp' ;

Starting restore at 09-JAN-13
allocated channel ORA_DISK_1
channel ORA_DISK_1: SID=1087 device type=DISK

channel ORA_DISK_1: restoring control file
channel ORA_DISK_1: restore complete, elapsed time: 00:00:07
output filename=/u02/app/oracle/oradata/cad01/control01.ctl
output filename=/u03/app/oracle/oradata/cad01/control02.ctl
Finished restore at 09-Jan-13
```

Auch hier kann man den Parameter `DB_NAME` nehmen, wenn die `DBID` vorher nicht gesetzt worden ist.

```
rman>restore controlfile from
 2>' /u04/app/oracle/flash_recovery_area/cad10/autobackup/2012_11_17/
 3> o1_mf_s_799558311_8bfxt0k.bkp' DB_NAME 'CAD10';
```

2.13.6 Restore Database

Nach dem herstellen der Control Files kann die Datenbank im mount Staus gebracht werden und die Datenbank Dateien werden wieder hergestellt. Mit der Option until logseq wird bis einem bestimmten Punkt zurück gesichert. Hierbei kann mit list archivlog all die vorhanden Archivelogs angezeigt werden, siehe 2.12.7.

```
rman>alter database mount;

database mounted
released channel: ORA_DISK_1

rman>restore database;

Starting restore at 09-JAN-13
Starting implicit crosscheck backup at 09-JAN-13
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=1086 device type=DISK
Crosschecked 4 objects
Finished implicit crosscheck backup at 09-JAN-13

Starting implicit crosscheck copy at 09-JAN-13
using channel ORA_DISK_1
Finished implicit crosscheck copy at 09-JAN-13

searching dor all Files in the recovery area
catalog files...
cataloging done

List of cataloged Files
=====
File Name: /u04/app/oracle/flash_recovery_area/cad10/autobackup/2012_11_17/
o1_mf_s_799558311_8bfxt0k.bkp

using channel ORA_DISK_1
channel ORA_DISK_1: starting datafile backup to restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup set
restoring datafile 0001 to /u02/app/oracle/oradata/cad10/system01.dbf
restoring datafile 0002 to /u02/app/oracle/oradata/cad10/sysaux01.dbf
restoring datafile 0003 to /u02/app/oracle/oradata/cad10/undotbs01.dbf
restoring datafile 0004 to /u02/app/oracle/oradata/cad10/users01.dbf
channel ORA_DISK_1: reading from backup piece
/u04/app/oracle/flash_recovery_area/cad10/backupset/2012_11_17/01_mf_nnnd0_
LEVEL0_8bffyqo_:bkp
channel                ORA_DISK_1:                piece
handel=/u04/app/oracle/flash_recovery_area/cad10/backupset/2012_11_17/01_mf
_nnnd0_ LEVEL0_8bffyqo_:bkp tag=LEVEL0
channel ORA_DISK_1: restore backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 01:23:34
Finished restore at 09-JAN-13
```

2.13.7 Recover Database

Vor dem öffnen der Datenbank muss sie `recoverd` werden. Möchte man nur das Recovern bis zu einer bestimmten Log Sequence durchführen, so hängt man die Option `until logseq` an den Befehl an.

```
rman>list archivelog all;
Key      Thrd Seq      S Low Time
-----
261691 1      121438 A 09-JAN-13
          File:    /u04/app/oracle/flash_recovery_area/cad10/archivelog/2013-01-
09/o1_mf_1_121438_8mgr2plp_.arc
261692 1      121439 A 09-JAN-13
          File:    /u04/app/oracle/flash_recovery_area/cad10/archivelog/2013-01-
09/o1_mf_1_121439_8mbtxpm_.arc

rman>recover database until logseq 121438;
```

Hier werden all Logfiles `recoverd`.

```
rman>recover database;

Starting recover at 09-JAN-13
using channel ORA_DISK_1

starting media recovery

channel ORA_DISK_1: starting archived log restore to default destination
channel ORA_DISK_1: restoring archive log
archived log thread=1 sequence=121438
channel ORA_DISK_1: restoring archive log
archived log thread=1 sequence=121439
channel ORA_DISK_1: reading from backup piece
/u04/app/oracle/flash_recovery_area/cad10/backupset/2012_11_17/01_mf_anmnn_
TAG201222217T002659_8bfplon3_.bkp
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:01:45
archived log file name=
/u04/app/oracle/flash_recovery_area/cad10/archivelog/2012_11_17/01_mf_1_121
438_8gwqs6fy_.arc thread=1 sequence=12438
channel default: deleting archived log(s)
unable to find archived log
archived log thread=1 sequence=121440
RMAN-00571: =====
RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS =====
RMAN-00571: =====
RMAN-03002: failure of recover command at 01/09/2013 14:31:34
RMAN-06054: media recovery requesting unknown archived log for thread 1
with sequence 121440 and starting SCN of 2893688957
```

Keine Panik bei der oben angezeigten Fehler Meldung. Diese Meldung besagt, das es keine weiteren Archived Logs gibt zum `recovern`. Die Datenbank muss nun mit der Option `resetlogs` geöffnet werden.

```
rman>alter database open resetlogs;

database opened
```

2.13.8 Restarbeiten

Damit man sich an die DB von einem Client anmelden kann, muss noch eine Passwort Datei erstellt werden.

```
oracle@woby1002>cd $ORACLE_HOME/dbs  
oracle@woby1002>orapwd file=orapwcad10 password=<passwd> entries=5
```

2.14 Online Backup (Cold Backup)

Ein Online Backup, kopieren der Dateien in ein anderes Verzeichnis, kann folgendermaßen gemacht werden.

2.14.1 Einzelne Tablespaces

```
sql>alter tablespace <tablespace_name> begin backup;  
oracle@woby1002>cp /oracle/oradata/cad10/uws01.dbf /backup/cad10/uws01.dbf  
sql>alter tablespace <tablespace_name> end backup;
```

2.14.2 Ganze Datenbank

```
sql>alter database begin backup;  
oracle@woby1002>cp /oracle/oradata/cad10/*.* /backup/cad10/*.*  
sql>alter database end backup;
```

3. Datapump (ab 10G)

3.1 Export

3.1.1 Datapump Tabellen / Views

Für das überwachen / kontrollieren gibt es verschiedenen Tabellen und Views.

<u>Tabelle / View</u>	<u>Beschreibung</u>
dba_datapump_jobs	Datapump Jobs.
dba_datapump_sessions	Datapump Sessions pro Job.
dba_export_objects	Liste der Exclude / Include Objecte.
Kupc\$datapump_quetab	Datapump Queue

3.1.2 Vorarbeiten

Als sys sich an die Datenbank anmelden und ein Exportverzeichnis der Datenbank bekannt geben. Das anzugebene Verzeichnis muss vorhanden sein.

```
sql>grant create any directory to <oracle_user>;
sql>create or replace directory <name> as '/transfer/dmp';
sql>grant read, write on directory <name> to <oracle_user>;
```

Erstellte Directories werden in der Tabelle dba_directories verwaltet.

```
sql>select * from dba_directories;
OWNER DIRECTORY_NAME    DIRECTORY_PATH
-----
SYS    DATAPUMP_DIR           /var/export
```

Directories werden mittel des Befehls drop gelöscht.

```
sql>drop directory <name>
```

3.1.3 Hilfe anzeigen

Die Hilfe zu Oracle Datapump wird mit der Option help=y angezeigt.

```
oracle@woby1002>expdp help=y
oracle@woby1002>impdp help=y
```

3.1.4 Export Tabellen

In dem nachfolgenden Beispiel sollen zwei Tabellen exportiert werden. Vorher sollte ein vorhandenes Export File gelöscht werden.

```
oracle@woby1002>expdp hr/<passwd>@cad01 tables=emp,dept directory=<name>
dumpfile=emp_dept.dmp logfile=exp_emp_dept.log
```


3.1.5 Export Schemas

Das zu erstellende Export File darf nicht existieren, sonst gibt es eine Fehlermeldung. Das Export File muss vorher gelöscht werden.

```
oracle@woby1002>expdp hr/<passwd>@cad01 schemas=hr directory=<name>
dumpfile=hr.dmp logfile=exp_hr.log
```

Für den Export des Schemas kann man die Parameter `include` oder `exclude` angeben.

```
oracle@woby1002>expdp system/<passwd>@cad01 schemas=hr include=table:"IN
('emp', 'dept')" directory=<name> dumpfile=hr.dmp logfile=exp_hr.log

oracle@woby1002>expdp system/<passwd>@cad01 schemas=hr exclude=table:"=
'bonus'" directory=<name> dumpfile=hr.dmp logfile=exp_hr.log
```

3.1.6 Export Datenbank

```
oracle@woby1002>expdp system/<passwd>@cad01 full=y directory=<name>
dumpfile=cad01.dmp logfile=exp_cad01.log

oracle@woby1002>expdp system/<passwd>@cad01 full=y include=grant
include=index directory=<name> dumpfile=cad01_full.dmp
logfile=cad01_full.log
```

3.1.7 Parameter Datei

Alle Parameter können auch in einer Datei stehen, die dann für einen Export / Import job aufgerufen wird.

```
oracle@woby1002>cat fullexport.par
userid='sys/<passwd>@<oracle_sid> as sysdba'
directory=data_pump_dir
dumpfile=exp_datapump.dmp
logfile=exp_datapump.log
schemas=cad01
exclude=table:"in (
'city',
'contact')",
Index: "in (
'city_n',
'contact_n')"
status=60
```

Mit dem Parameter `status=60` wird bei jedem Export ausführliche Meldungen ausgegeben. Die Angabe erfolgt in Sekunden.

3.1.8 Export Query

```
query:<table>:"where department_id >20 and salasry > 1000"
```

Es werden nur die Daten aus der angegebenen Tabelle exportiert, auf denen der Filter passt. Bei einem Windows System sind zwei / drei Hochkommentars (") zu setzen. Zwischen den Hochkommentars muss ein Leerzeichen sich befinden. Siehe auch 3.2.6.

```
query=""where zeit < to_date('17.01.2012','DD.MM.YYYY hh24:mi:ss') ""
```

In dem obigen Beispiel werden nur die Daten exportiert, die vor dem angegebenen Datum sind.

3.1.9 Datapump Jobs löschen

Alle Datapump Jobs werden in der Tabelle `dba_datapump_jobs` verwaltet. Das Löschen eines Jobs geschieht mit dem `drop table <owner>.<job_name>` Befehl.

```
sql>select * from dba_datapump_jobs where state='EXECUTING';
OWNER JOB_NAME                OPERATION  JOB_MODE  STATE        DEGREE
-----
sys   sys_export_schema_02  export    full      executing    1

sql>drop table sys.sys_export_schema_02 purge;
Table dropped.
```

3.1.10 Parameter

Die Angabe des Directories kann nicht nur über den Parameter `directory` erfolgen, sondern auch direkt den Parametern `dumpfile` und `logfile`. Vor der Angabe der Datei, wird der Directory Name angegeben und als Abschluß erfolgt ein Doppelpunkt.

```
oracle@woby1002>expdp hr/<passwd>@cad01 schemas=hr dumpfile=exp_path:hr.dmp
logfile=exp_path:exp_hr.log
```

3.1.11 Dump Files splitten

Möchte man mehrere Dump Files mit einer bestimmten Größe erstellen, so gibt man im Dateinamen ein `%u` mit an. Mit dem Parameter `filesize` wird die Größe der Dump Dateien angegeben. Die Angabe der Einheit erfolgt mit `B|K|M|G`. Sollen mehr als 99 Dateien erstellt werden, so gibt man mehrere Dump File Namen an.

```
dumpfile=exp_cad10_full_%u.dmp filesize=20M
dumpfile=exp_cad10_full_%u.dmp,exp_cad10_full_1%u.dmp filesize=2M
```

3.2 Import

3.2.1 Import Tabellen

```
oracle@woby1002>impdp hr/<passwd>@cad01 tables=emp,dept directory=<name>
dumpfile=emp_dept.dmp logfile=imp_emp_dept.log
```

3.2.2 Import Schemas

```
oracle@woby1002>impdp hr/<passwd>@cad01 schemas=hr directory=<name>
dumpfile=hr.dmp logfile=imp_hr.log
```

Für den Import des Schemas kann man die Parameter include oder exclude angeben.

```
oracle@woby1002>impdp system/<passwd>@cad01 schemas=hr include=table:"IN
('emp', 'dept')" directory=<name> dumpfile=hr.dmp logfile=imp_hr.log
```

```
oracle@woby1002>impdp system/<passwd>@cad01 schemas=hr exclude=table:"=
'bonus'" directory=<name> dumpfile=hr.dmp logfile=imp_hr.log
```

3.2.3 Import Datenbank

```
oracle@woby1002>impdp system/<passwd>@cad01 full=y directory=<name>
dumpfile=cad01.dmp logfile=imp_cad01.log
```

3.2.4 Parameter Datei

Anstelle der ganzen Syntax in der Console einzugeben, kann man auch eine Parameter Datei erstellen und diese dann mit der Export oder Import Funktion aufrufen.

Beispiel eines Par-Files:

```
directory=hr_dmp
schemas=hr
dumpfile=hr.dmp
logfile=hr.log

oracle@woby1002>expdp hr/<passwd>@cad01 parfile=hr_schema.par
```

3.2.5 Remap Data

Soll z.B. der exportierte Tablespace in einen anderen Tablespace importiert werden, so gibt es hierfür die Option REMAP_TABLESPACE. Um Datafiles umzubenennen, wird die Option REMAP_DATAFILES angegeben. Mit der Option REMAP_SCHEMA kann auch das Schema umbenannt werden. Diese Optionen können natürlich auch in dem Parameter File eingetragen werden.

```
oracle@woby1002>impdp hr/<passwd>@cad01 remap_tablespace=tbs_1:tbs_6
directory=hr_dmp dumpfile=hr.dmp logfile=imp_hr.log
```

```
oracle@woby1002>impdp hr/<passwd>@cad01
remap_datafile="'c:\db1\hrdata\tbs6.dbf': '/db1/hrdata/tbs7.dbf' "
directory=hr_dmp dumpfile=hr.dmp logfile=imp_hr.log
```

```
oracle@woby1002>impdp hr/<passwd>@cad01 directory=hr_dmp dumpfile=imphr.log  
tables=hr.emp remap_schema=hr:hr1
```

3.2.6 Query Abfrage

Soll aus einem kompletten Export nur eine Teilmenge importiert werden, so kann man hierzu eine Query Abfrage erstellen. Eine Query Abfrage kann auch für einen Export genommen werden. Die Syntax für eine Query Abfrage ist: <schemaName>.<TableName>:"<Query>".

```
oracle@woby1002>cat hr_schema.par  
directory=hr_dmp  
dumpfile=hr_query.dmp  
logfile=hr_query.log  
table=hr  
query=employees:\ "where salary>'1000'" \ "
```

3.2.7 Tabelle vorhanden

Ist eine Tabelle in der Datenbank vorhanden, so kann man die mit der Angabe der Option `table_exist_actions` angeben, was mit dem Import geschehen werden soll. Als Werte können `Skip`, `Append`, `Truncate` oder `Replace` angegeben werden.

```
oracle@woby1002>cat hr_schema.par  
directory=hr_dmp  
dumpfile=hr_query.dmp  
logfile=hr_query.log  
table=hr  
table_exist_actions=append
```

4. Export / Import (bis 9i)

4.1 Export

Einen Export der Datenbank oder auch nur einer Tabelle kann man mit dem Kommandozeilen Befehl `exp` vornehmen.

```
C:\>exp cv3d/cv3d tables="(ddd_collaboration_data_info)"  
file=/u01/data/backup/full.dmp log=/u01/data/backup/full.log
```

Alles von einem Oracle User exportieren.

```
C:\>exp cv3d/cv3d file=d:\oradmp\cv3d.dmp log=d:\oradmp\cv3d.log
```

Anzeige der Hilfe.

```
C:\exp help=y
```

4.2 Import

Einen Import einer Sicherung wird auf der Kommandozeile mit dem Befehl `imp` vorgenommen. Möchte man eine Tabelle von einem anderen Oracle User Importieren, so muss die Option `fromuser/touser` angegeben werden.

```
C:\>imp sys/<passwd> file=/u01/data/backup/full.dmp fromuser=cv3d  
touser=cv3dbck tables="(ddd_collaboration_data_info)"
```

Anzeige der Hilfe.

```
C:\imp help=y
```

5. Flashback

5.1 Papierkorb

5.1.1 Inhalt des Papierkorbs anzeigen

```
sql>show recyclebin;
```

5.1.2 Inhalt des Papierkorbs löschen

```
sql>purge recyclebin;
```

5.2 Database / Tabelle

5.2.1 Table zurücksetzen

```
sql>flashback table <table_name> to before drop;
```

Bevor man eine Tabelle mit den nachfolgenden Befehl zurücksetzen kann, muss für die Tabelle `row movement` eingeschaltet sein.

```
sql>alter table mitarbeiter enable row movement;

sql>select first_name, last_name from mitarbeiter as of timestamp
2>(systimestamp - interval '1' hour) where last_name = 'Ellison';

sql>flashback table mitarbeiter to timestamp
2>(systimestamp - interval '1' hour)
```

Ausser der Angabe von `hours` gibt es noch `minute` und `second`. Anstelle einer Zeitabfrage kann auch eine `SCN` Abfrage erfolgen.

5.2.2 Database zurücksetzen

Um eine Datenbank zurücksetzen zu können, muss man ausreichende Berechtigungen haben.

```
sql>flashback database to timestamp (systimestamp - interval '5' minute);
```

6. SQL Loader

Mit Hilfe des SQL Loaders kann man Daten in einer Tabelle importieren.

6.1 Aufruf

Der SQL Loader wird mit `sqlldr <parameter1> <parameter2> ...` aufgerufen.

Folgende Parameter können gesetzt werden.

<u>Parameter</u>	<u>Beschreibung</u>
<code>userid=<connect_string></code>	Oracle User
<code>control=<control_datei></code>	Steuerdatei
<code>log=<log_datei></code>	Log Datei, optional
<code>bad=<bad_datei></code>	Bad Datei, optional
<code>discard=<discard_datei></code>	Discard Datei, optional
<code>discardmax=n</code>	Nach n Datensätzen im Discard File wird der Import abgebrochen.
<code>skip=n</code>	Die ersten n Datensätze nicht importieren.
<code>load=n</code>	Laden von n Datensätzen.
<code>errors=n</code>	Abbrechen des Imports bei n Fehlerhaften Datensätze.
<code>direct=true</code>	True nur auf dem Server, der Cache Speicher der Instanz wird dann umgangen, der Import läuft schneller.

6.2 Beispiele

Anstelle von `Append` kann man auch `insert`, `replace` oder `truncate` nehmen. Bei `replace` werden vorher alle Daten aus der Tabelle gelöscht.

```
oracle@woby1002>cat LoadDaten1.cnf
LOAD DATA
INFILE 'mess1.dat'
APPEND
INTO TABLE MESS_VALUES
FIELDS TERMINATED BY ';'
(KKS,
 ZEIT DATE "DD.MM.YYYY HH24:MI:SS",
 VALUE)

oracle@woby1002>cat mess1.dat
15BQ001MS001;13.03.2014 09:12:05;1.7654
15BQ001MS001;13.03.2014 09:12:15;1.8453
15BQ001MS001;13.03.2014 09:12:25;1.6239
15BQ001MS001;13.03.2014 09:12:35;1.7129

oracle@woby1002>sqlldr userid=uws/uws log=imp_uws.log
control=LoadDaten1.cnf
```

Sind die Werte mit Hochkommas versehen, so gibt man den Parameter `OPTIONALY ENCLOSED BY` mit an.

```
oracle@woby1002>cat LoadDaten2.cnf
LOAD DATA
INFILE 'mess2.dat'
APPEND
INTO TABLE MESS_VALUES
FIELDS TERMINATED BY ';' OPTINONALY ENCLOSED BY '"'
(KKS,
 ZEIT DATE "DD.MM.YYYY HH24:MI:SS",
 VALUE)

oracle@woby1002>cat mess2.dat
"15BQ001MS002";13.03.2014 09:16:03;1.7654
"15BQ001MS002";13.03.2014 09:16:13;1.8453
"15BQ001MS002";13.03.2014 09:16:23;1.6239
"15BQ001MS002";13.03.2014 09:16:33;1.7129
```

Mann kann die Daten auch in der Control Datei vorhalten. Gibt es bei den Daten am Ende auch Einträge ohne Werte, so werden sie mit dem Parameter `TRAILING NULLCOLS` als `NULL` Werte angenommen, andernfalls wird die Zeile als `BAD` deklariert.

In dem unterem Beispiel wird das Feld `DEPTNO` mit einem Doppelpunk beendet und das Feld `SEQ` wird automatisch gefüllt.

```
oracle@woby1002>cat LoadDaten3.cnf
LOAD DATA
INFILE *
APPEND
INTO TABLE CONTACT
FIELDS TERMINATED BY ',' OPTINONALY ENCLOSED BY '"' TRAILING NULLCOLS
(ID,
 FIRST_NAME,
 LAST_NAME,
 CITY,
 STREET,
 DEPTNO CHAR TERMINATED BY ':',
 SEQ(MAX,1) )

BEGINDATA
100, "Hans", "Meiser", "Köln", "Amselweg", 10:201
101, "Anton", "Meise", "Bonn", "Waldweg", 20:300
102, "Karl", "May", "Segeberg", "Wild-West Strasse", 32:501
103, "Rudi", "Rabe", "Bochum", "Grotten Weg", 44:601
```


In dem nächsten Beispiel sind die Werte nicht mit einem Separator getrennt, sondern in einer festen Struktur.

```
oracle@woby1002>cat LoadDaten4.cnf
LOAD DATA
INFILE 'contact.dat'
APPEND
INTO TABLE CONTACT
(
ID          POSITION(01:03)  INTEGER EXTERNAL,
FIRST_NAME  POSITION(05:09)  CHAR,
LAST_NAME   POSITION(11:16)  CHAR,
CITY        POSITION(18:25)  CHAR,
STREET      POSITION(27:43)  CHAR,
DEPTNO      POSITION(45:46)  INTEGER EXTERNAL,
SAL         POSITION(48:54)  DECIMAL EXTERNAL
)

oracle@woby1002>cat contact.dat
100 Hans  Meiser Köln      Amselweg          10 1400.00
101 Anton Meise  Bonn      Waldweg           20  300.00
102 Karl   May      Segeberg Wild-West Strasse 32  850.00
103 Rudi   Rabe    Bochum   Grotten Weg       44 1650.00
```

7. Error & Erro Codes

7.1 RMAN-06820

In den Rman Log File steht folgende Meldung: Warning: failed to archive current ..

Solution:

Den Backup Job mit `rman target sys/<passwd>` anstelle von `rman target /` starten.

8. Copyright

Dieses Dokument ist urheberrechtlich geschützt. Das Copyright liegt bei Uwe Schimanski.

Das Dokument darf gemäß der GNU *General Public License* verbreitet werden. Insbesondere bedeutet dieses, daß der Text sowohl über elektronische wie auch physikalische Medien ohne die Zahlung von Lizenzgebühren verbreitet werden darf, solange dieser Copyright Hinweis nicht entfernt wird.