

Angewandte Datentechnik
Matura Themenausarbeitung

Backup & Recovery mit Oracle

Maximilian Riedl
maximilian.riedl@gmx.at
HTBLVA Spengergasse 20, 1050 Wien
2004/2005



V 1.1

Vorfeld

Grundsätzlich Überlegungen zum Thema Backup&Recovery sind schon im Vorfeld bei der Installation des Betriebssystems zu treffen:

- Verteilte Speicherorte auf verschiedenen Platten oder Partitionen für
 - User-Files
 - Log-Files
 - System-Files
 - Backup-Files
 - Testumgebung
- Mehrfache Datenhaltung (zB: RAID)
- Zugriffsbeschränkung für Gruppen und User
- Speicherplatz-Kontingent für Anwender

Zeitpunkt

Bei einem Backup sollten - wenn möglich - keine Transaktionen am System durchgeführt werden, um einen konsistenten Zustand der Backup-Daten zu gewährleisten. Daher kommt es auf die Frequentierung der Datenbank an:

Zugriff auf DB < 24/7

Es ist Möglich, das Backup dann durchzuführen, wenn keine Benutzer eingeloggt sind bzw. keine Transaktionen offen sind. In der Nacht kann das System für den Zeitpunkt der Sicherung gesperrt werden und ein konsistentes Backup kann erstellt werden.

Zugriff auf DB > 24/7

Sollten während eines Backups gerade Transaktionen durchgeführt werden, müssen unbedingt die Redo-Log-Files beim Wiederherstellen des Systems beachtet werden. Es kann also ein konsistentes System wiederhergestellt werden, wenn ein Backup vom derzeitigen (gerade benutzen) System gemacht wird und alle getätigten Transaktionen durch die Redo-Logs gespeichert werden. In Oracle ist es möglich, mit dem *Recover-Manager* und den Redo-Logs das System wieder herzustellen.

Backup-Methoden

Um sicher zu gehen, dass alle Daten bei einem möglichen Festplatten-Crash wieder hergestellt werden können, ist ein zyklisches Sichern der Daten anzuwenden:

Folgendes Szenario kann angewendet werden:

- am 1. eines Monats läuft ein komplettes Backup des Systems (**Level 0**)
 - es folgen darauf bis zum nächsten Samstag jeden Tag Level 2 Backups (**Level 2**) (Dieses sichert alles was sich seit dem letzten Level 1 Backup geändert hat)
 - Am ersten Sonntag folgt dann ein Level 1 Backup (**Level 1**) (Dieses sichert alles was sich seit dem letzten Level 0 Backup geändert hat)
- ➔ Die nächsten 6 Tage folgen wieder tägliche Level 2 Backups usw.....

Für ein Monat gesehen würde das folgendermaßen aussehen:

0 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2

Durch diese Methode ist es möglich, selbst am Ende eines Monats, das System schnell wieder herzustellen:

- Zuerst wird das Backup 0 eingespielt
- Im Anschluss das letzte Backup 1
- Danach das letzte Backup 2

Das System kann somit wieder auf den letzten Stand gebracht werden und es müssen nicht alle Backups in Folge eingespielt werden. Das spart Zeit und Speicherplatz!

Diese Backup-Methode kann am Server automatisiert ausgeführt werden. Unter Linux bietet sich eine Kombination von *CRON-Jobs* und dem Backup-Programm *dump* hervorragend an.

Export mit Oracle (exp)

Die Export Funktion von Oracle bietet die Möglichkeit, Tabellen mit ihren zugehörigen Indexen, Berechtigungen und Constraints in eine Datei zu exportieren. Die Datei hat ein eigenes Format und kann durch das Import-Tool von Oracle wieder gelesen werden, auch auf einem anderen Oracle-Datenbanksystem mit unterschiedlicher Hardwareumgebung.

Um einen Export durchführen zu können, sollte sichergestellt sein, dass:

- Ausreichend Speicherplatz am Backup-Medium vorhanden ist (ansonsten Fehlermeldung und Abbruch)
Um den benötigten Speicherplatz zu bestimmen, kann das DataDictionary verwendet werden:
`SELECT SUM(BYTES) FROM USER_SEGMENTS WHERE SEGMENT_TYPE='TABLE';`
- Benötigte Systemrechte (EXP_FULL_DATABASE) vorhanden sind

Der Aufruf des Exports kann folgendermaßen erfolgen:

- Über die Kommandozeile mit Parametern
 - `exp username/password@host PARAMETER=(value1,value2,...,valuen)`
- Interaktiv: Benutzer wird aufgefordert benötigte Parameter einzugeben
 - `exp username/password@host`
- Durch Parameter-Dateien: Die Parameter können in einer Datei einfach angegeben und bearbeitet werden:
 - `exp username/password@host PARFILE=filename`
 Beispiel für eine Parameter-Datei:

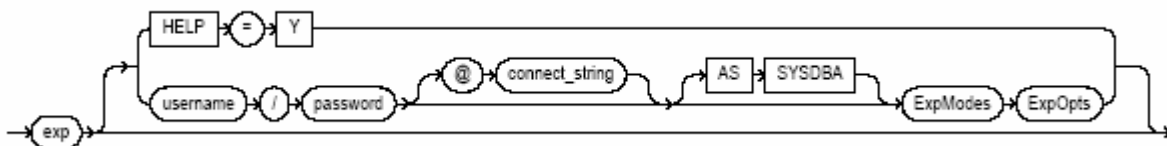

```
FULL=y
FILE=dba.imp
GRANTS=y
INDEXES=y
CONSISTENT=y
```

Der Export bietet vier verschiedene Möglichkeiten:

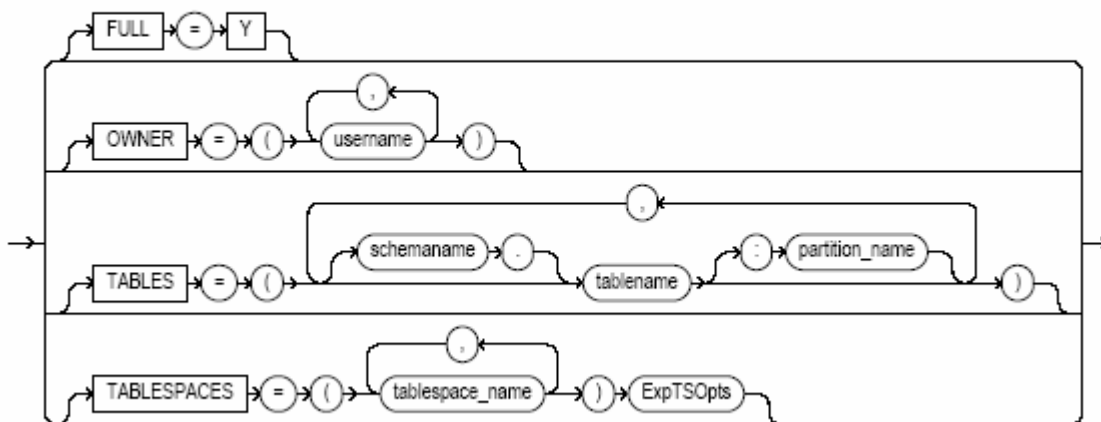
- Full
 - User (Owner)
 - Table
 - Tablespace
- } Normale User } Administrator

Die genauen Unterschiede sind sehr umfangreich und können im Oracle-Handbuch eingesehen werden.

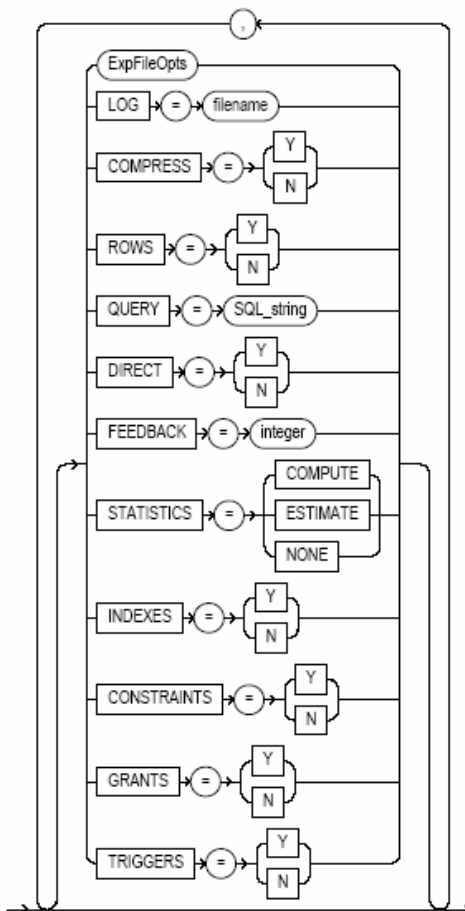
Der Syntax von exp sieht folgendermaßen aus (siehe Oracle-Handbuch)



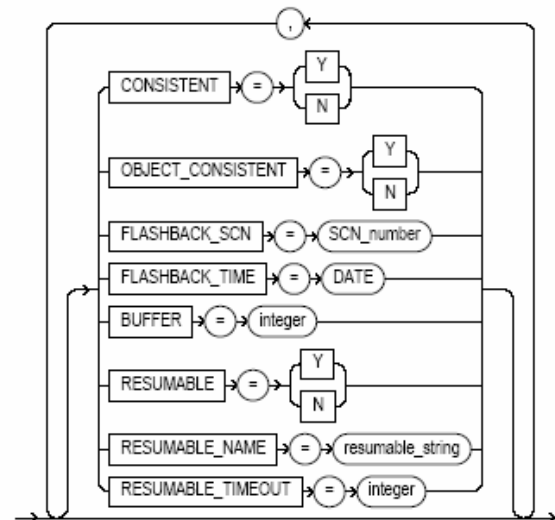
ExpModes



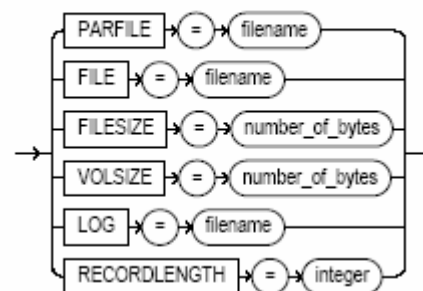
ExpOpts



ExpOpts_continued



ExpFileOpts



Compress: bietet die Möglichkeit, die Daten zu zu komprimieren um Speicherplatz zu sparen. Sollte immer verwendet werden und ist standardmäßig eingeschalten.

Consistent: Durch diesen Paramter ist es möglich, einen konsistenten Export zu erzeugen, auch wenn Transaktionen durchgeführt werden. Der Export dauert jedoch sehr lange und ist zeitaufwendig und ist erst zuende, wenn es die Transaktionen erlauben.

Constraints: exportiert alle Table_Constraints

Grants: exportiert in FULL-Mode alle Grants, in USER-Mode nur die der eigenen Tabellen.

File: gibt die Datei an in welche Exportiert werden soll.

Rows: ob die Zeilen mitexportiert werden sollen (Standardmäßig eingeschalten)

Tables: gibt an, welche Tabellen exportiert werden sollen -> Alle = *

Beispiel für exp im User-Mode:

Der Export im User-Mode bietet sich für alle Datenbankbenutzer an, um ihre persönlichen Tabellen zu sichern. Dies liegt sicherlich auch in der Verantwortung des DBA, jedoch kann eine selbstständige Sicherung der Daten benötigt werden. (auch für Transport, etc...)

Parameter File Methode

```
> exp max/mamut PARFILE=parameter.dat
```

parameter.dat beinhaltet folgende Informationen:

```
FILE=max.dmp  
OWNER=max  
GRANTS=y  
ROWS=y  
COMPRESS=y  
LOG=log.txt  
FEEDBACK=1
```

Command-Line Methode

```
> exp max/mamut FILE=max.dmp OWNER=max GRANTS=y ROWS=y COMPRESS=y
```

In diesem Beispiel werden alle Objekte des Benutzers *max* exportiert und komprimiert.

Import mit Oracle (imp)

Das Import-Tool liest die Daten aus der mit exp erstellten Datei und importiert diese in eine Oracle Datenbank. Zuerst werden die Tabellen angelegt, danach die Daten importiert und die Indexe gebildet. Im Anschluss werden Trigger und Constraints ausgeführt bzw. angelegt.

Für einen FULL-Import wird das CREATE SESSION bzw. das IMP_FULL_DATABASE Privileg benötigt, welches ein DBA besitzt. Das Import-Tool bietet die äquivalenten Möglichkeiten wie der Export und kann entweder per Parameter-Datei oder Interaktiv ausgeführt werden.

Wenn in bereits bestehende Tabellen importiert werden soll, kann es zu Fehlermeldungen kommen aufgrund von „falsch“ ausgeführten CREATE-STATEMENTS (da die Tabelle ja bereits besteht) und möglicher referenzieller Integrität zu anderen Tabellen. Um solchen Fehlern entgegen zu wirken, kann die referenzielle Integrität beim Import deaktiviert werden, indem der Parameter IGNORE=y mitgegeben wird. **Vorsicht**, sollten Tabellen bereits bestehen, werden die Datenzeilen nochmals eingetragen was zu doppelten Einträgen führt!

Beispiel für imp für ausgewählte Tables und User:Parameter File Methode

```
> imp SYSTEM/password PARFILE=parameter.dat
```

parameter.dat beinhaltet folgende Informationen:

```
FILE=dba.dmp  
SHOW=n  
IGNORE=n  
GRANTS=y  
FROMUSER=max  
TABLES=(dept,emp)
```

Command-Line Methode

```
> imp SYSTEM/password FILE=dba.dmp FROMUSER=max TABLES=(dept,emp)
```

Beispiel für imp für anderen User:Parameter File Methode

```
> imp SYSTEM/password PARFILE=parameter.dat
```

parameter.dat beinhaltet folgende Informationen:

```
FILE=max.dmp  
SHOW=n  
IGNORE=n  
GRANTS=y  
ROWS=y  
FROMUSER=max  
TOUSER=franki  
TABLES=(unit,manager)
```

Command-Line Methode

```
> imp SYSTEM/password FROMUSER=max TOUSER=franki FILE=max.dmp -  
TABLES=(unit,manager)
```

Ignore: Der Import überspringt Fehlermeldungen beim Erzeugen von Datenbankobjekten, Datenzeilen können in schon bestehenden Tabellen doppelt eingetragen werden.

Constraints: kann für den Import auf n gesetzt werden um Fehler zu vermeiden

Fromuser: welche Schemen importiert werden sollen (wichtig bei IMP_FULL_DATABASE)

Touser: In welchen User-Workspace importiert werden soll

Show: in diesem Fall wird der Inhalt der export-Datei auf dem Bildschirm ausgegeben, es werde jedoch keine Befehle ausgeführt.