

# Das Leben mit dem (Datenbank) Klon

**Christian Pfundtner  
DB Masters GmbH**

**Stammersdorfer Str. 463  
2201 Gerasdorf  
Österreich**

## **Schlüsselworte**

Datenbank, Klon, Clone, Storage, Backup, Lösung

## **Einleitung**

Genaugenommen bräuchten wir alle einen (oder mehrere) Klone – nur gibt es viele Gründe, warum wir keine nutzen. Dieser Vortrag soll Ihnen angefangen mit Anwendungsbeispielen, über die Erklärung was eigentlich ein Datenbank-Klon ist bis hin zum „Leben mit dem Klon“ die wichtigsten Informationen zu dem Thema vermitteln.

Die Präsentation ist auf Wunsch der DOAG in englischer Sprache verfasst, das Manuskript habe ich absichtlich auf deutsch gelassen, da die meisten der Leser vermutlich aus DACH kommen.

## **Warum brauchen wir einen Klon?**

Nun, der klassische Ansatz für den man einen Klon sehr gut verwenden kann ist natürlich, dass man verschiedene Umgebungen wie Produktion, Test, Entwicklung, Schulung, etc. nutzt. In den meisten Fällen werden solche „Umgebungen“ durch Kopien voneinander erzeugt und dann für verschiedenlange Zeiträume genutzt. In den meisten Fällen werden diese Umgebungen durch ein (RMAN) DUPLCIATE bzw. ein Restore aus einer anderen Umgebung erstellt.

Ein aus meiner Sicht viel wichtigerer Anwendungsfall ist das Testen von Upgrades – egal ob Oracle Patchsets oder Releaseupgrades oder auch Applikationsupgrades.

Speziell in Umgebungen, wo die Entwickler am offenen Herzen (auf Produktionsdatenbanken) Zugriff haben, ist es schon öfter passiert, dass „Ooops, das war die falsche Tabelle!“. Gemeint ist damit, dass die Daten oder die ganze Tabelle versehentlich zerstört wurden. Hier bietet Oracle zwar mit verschiedenen Flashbacktechnologien die Möglichkeit den Fehler in vielen Fällen zu reparieren. Die Voraussetzung dafür ist aber, dass man entsprechen vorgesorgt hat (UNDO\_RETENTION, Flashback Logs, etc).

Ein weiteres Anwendungsbeispiel ist das „Offloading“ von Tätigkeiten von der Primärdatenbank. Sei es Reporting, Backup oder aber auch das Erzeugen von Optimiser Statistiken. Für einige dieser Anforderungen kann man den kostenpflichtige Oracle Active Data Guard nutzen, aber nicht in jedem Fall. Spätestens aber mit dem Erzeugen von Optimiser Statistiken geht das nicht mehr. Zusätzlich ist zu bedenken, dass viele Kunden nur die Oracle Standard Edition Datenbank lizenziert haben – hier ist Data Guard überhaupt nicht möglich.

## **Was ist ein Klon?**

Für viele ist ein Klon einfach nur eine 1:1 Kopie. Näher betrachtet, muss man diese Definition allerdings auf „Ein Klon ist eine Kopie der Datenbank von einem Zeitpunkt, den ich mir aussuchen kann“ erweitern. Das bedeutet aber auch, dass viele klassische Ansätze (RMAN Duplicate oder Restore) für diese Definition nicht (gut genug) funktionieren. Klone erzeugt man viel besser und sinnvoller mit Storage Integration oder notfalls auch Softwarelösungen wie ACFS oder ähnliches.

## **Erzeugen eines Klon´s**

Klone erzeugt man am besten auf der Basis von (Storage oder Software) Snapshots. Diese Snapshots bieten eine 1:1 Version der Datenbank zu dem Zeitpunkt an dem der Snapshot erzeugt wurde. Wenn man sich jetzt beispielsweise jeden Tag zumindest einen Snapshot erzeugt und diese dann für einen längeren Zeitraum – beispielsweise ein Monat – verfügbar lässt, kann die Datenbank (mit Hilfe der Architelogfiles) sehr schnell und einfach auf jeden beliebigen Zeitpunkt aus Klone erzeugen indem man den Snapshot als Read/Write Kopie verfügbar macht und die benötigten Archivelog Files aufrecovered. Im Gegensatz zu einer klassischen Lösung (Restore) dauert das Read/Write zur Verfügung stellen eines Snapshots meist nur Sekunden bis wenige Minuten. Einige Storage Lösungen (wie beispielsweise NetApp) bieten bis zu 255 Snapshots an und durch die Integration von Secondary Storages mittels SnapVault, kann man auf der Primären Storage ruhig mit stündlichen Snapshots arbeiten, damit man für einen Kone zu einem bestimmten Zeitpunkt innerhalb der letzten 7-10 tage noch schneller kommen kann. Mit Hilfe der Secondary Storages mit SnapVault (hier würde man dann nur einen Snapshot pro Tag aufheben), kann man dann ganz locker 6-9 Monate in die Vergangenheit gehen.

Natürlich gibt es auch Software Lösungen, die Snapshot Technologien anbieten, wie Oracle dNFS, ACFS, ZFS, etc. diese haben aber alle die verschiedensten Limitierungen im Vergleich zu einer Storage basierten Lösung. Die aus meiner Sicht größte Limitierung ist, dass man dafür in den meisten Fällen die CPU des Datenbankservers „missbraucht“ was zu höheren Oracle Lizenzkosten führt, weil man mehr CPU Cores lizensieren muss.

## **Das Leben mit dem Klon**

Je nach Storageanbieter gibt es verschiedenste Tools und Lösungen um einen Snapshot auf Storagebasis zu erzeugen. Die meisten Hersteller haben zumindest:

- Eine GUI Lösung
- Eine Command Line Lösung
- Ein API bzw. eine Scripting Anbindung

Durch die verschiedenen Tools und Werkzeuge (GUI, Command Line, API) kann man einerseits die Automatisierung (Snapshots aber auch Klonerstellung) nutzen, aber auch onDemand – Beispielsweise ein DBA der keine Klon für ein Datenbankupgradetest benötigt – und Self Service Lösungen, wo sich ein Entwickler für einen Test schnell mal einen aktuellen Klon einer Datenbank erzeugen kann, anbieten.

Das wichtigste ist, dass man sich auch überlegt, wie lange man einen Klon nutzen möchte. Bei typischen Datenbanken kann man von folgenden zusätzlichen Platzbedarf auf der Storage ausgehen (Zusätzlich zum Platz den die Produktionsdatenbank sowie die Archive Log Files benötigen):

1 Tag	einige weniger %
1 Woche	typischerweise 5-10%
1 Monat	typischerweise 20-25%
1 Quartal	typischerweise 20-50%

Das bedeutet, dass man für eine Produktionsdatenbank und einen Klon, den man bis zu einem Monat nutzen möchte in Summe nur ca. 25% mehr Platz benötigt als die Produktionsdatenbank alleine. Wenn man mit klassischen Methoden (Backup und Restore) arbeitet, braucht man mehr wie den doppelten Platz → Klon sparen daher Platz!

Das hat selbst SAP erkannt und bietet mit dem SAP ILM eine Lösung in der die verschiedenen Umgebungen (Produktion, Entwicklung, Test, Schulung, QA,...) basierend auf Datenbank Klone bereitgestellt wird. Der Vorteil ist, dass man in Summe nur einen Bruchteil des Storagebedarfs hat, wie bei einer klassischen Lösung. Das Erzeugen einer Umgebung geht ebenfalls viel schneller als der „alte“ Weg.

### **Die Implementierung**

Wie helfen uns jetzt diese Klone auf Snapshot Basis im täglichen Leben?

Nun, wenn man in einer Datenbank mehrere Applikationen betreibt, gibt es immer mal wieder die Anforderung diese Applikationen auf zwei (oder mehr) Datenbanken zu verteilen, weil die Last zu groß geworden ist, oder es andere Themen bei der Koexistenz der Applikationen gibt (Beispielsweise zwingt ein Hersteller zum Upgrade auf eine Version, die der andere Hersteller nicht unterstützt, etc).

Hier kann man wie folgt vorgehen:

- Stoppen der Applikationen, die mit Hilfe des Klons abgesplittet werden.
- Erzeugen eines Snapshots und daraus sofort einen Klon
- Applikationen, die auf dem „Original“ laufen, haben diesen Vorgang nichts mitbekommen.
- Applikationen, die auf dem Klon betrieben werden sollen, brauchen jetzt einen neuen CONNECT String und können dann sofort gestartet werden.
- Optional: Den Klon mit Storgemitteln „absplitten“ (zu einem Original machen)
- In den beiden Datenbanken, die jeweils nicht mehr benötigten Schematas und Tabellen dropen.

Downtime: im Minutenbereich – und auch das nur für jene Applikationen, die abgesplittet werden.

Natürlich funktioniert das auch für den Klassiker schlecht hin: Den Klon zum Testen von Upgrades (Oracle aber auch Applikationsupgrades). Einfach einen Klon – idealerweise auf einem anderen Rechner – erzeugen und Upgrade testen.

Auch das versehentliche Datenverstümmeln oder Tabellen löschen hat seinen Schrecken verloren, einfach einen Klon zu einem Zeitpunkt erzeugen wo alles noch OK war und die Daten entsprechen exportieren oder anderweitig übernehmen.

Wenn man einen Klon zum Reduzieren der Last auf dem Produktionssystem nutzt, gibt es viele Anwendungsmöglichkeiten, beispielsweise:

Man Erzeugt man einen Klon auf einem anderen Server und

- führt dort das Backup durch
- Lässt dort die langlaufenden Auswertungen laufen
- Erzeugt hier die Optimizer Statistiken und importiert diese auf das Produktionssystem zurück
- Nutzt den Klon für ETL um das DWH zu aktualisieren
- Etc

### **Zusammenfassung**

Sie sehen, es gibt viele Anwendungsbeispiele wo ein Klon hilfreich eingesetzt werden kann. Klon kann man sowohl im SAN als auch im NAS Bereich erzeugen, wobei das NAS Bereich deutlich flexibler und einfacher zu nutzen ist wie SAN Lösungen.

Unsere Empfehlung: „Go for Clone“

### **Kontaktadresse:**

Christian Pfundtner

DB Masters GmbH

Stammersdorferstr. 463  
2201 Gerasdorf  
Österreich

eMail: [Christian.Pfundtner@dbmasters.at](mailto:Christian.Pfundtner@dbmasters.at)

Tel: +43 699 15037884

Internet: [www.dbmasters.at](http://www.dbmasters.at)