

Delphix bietet daher eine Möglichkeit an, die Daten auf dem Weg zum Klon zu maskieren. Bei CloneDB und Snapshot Clone sind dem Autor solche Verfahren nicht bekannt. Als Alternative bietet sich hier an, eine Kopie der Datenbank zu erstellen, die mithilfe der Oracle-Data-Masking-Option gesäubert und dann als Quelle für die Klonen genutzt wird.

Fazit

Generell können die genannten Verfahren den Storage-Bedarf von Datenbank-Kopien drastisch reduzieren. Delphix spricht davon, dass nur noch 5 Prozent der ursprünglichen Ressourcen erforderlich sind. Das ist sicherlich in einigen Um-

gebungen ein durchaus realistischer Wert. Allerdings eignen sich die hier vorgestellten Verfahren nicht für Performance-Tests, da der zusätzliche I/O die Messungen verfälschen würde.

Die Multitenant-Option eröffnet sicherlich noch weitere Möglichkeiten des Cloning, allerdings beschränkt sich dieses Verfahren auf Datenbanken, die auf einem gemeinsamen Server (natürlich auch als RAC) betrieben werden. Die anderen vorgestellten Lösungen sind mit den aktuellen Oracle-Versionen einsetzbar und können in Zukunft helfen, schneller, einfacher und vor allen Dingen in größerem Maße Datenbanken für Test und Entwicklung zur Verfügung zu stellen.



Johannes Ahrends
johannes.ahrends@carajandb.com

Schnelles Datenbank-Cloning mit Snapshots – ein Überblick

Manuel Hoßfeld, ORACLE Deutschland B.V. & Co. KG

Inhaltlich identische Kopien („Klone“) von Datenbanken sind für verschiedene Zwecke praktisch oder sogar notwendig – sei es als Kopie einer Produktionsdatenbank für Entwicklungszwecke oder als Basis von Tests, die man nicht an einer Produktionsdatenbank durchführen kann oder will.

Normalerweise erfordert das Anlegen eines Datenbank-Klons das vollständige Kopieren aller zum Original gehörenden Datenbank-Dateien. Es gibt inzwischen jedoch verschiedene alternative Möglichkeiten, Datenbanken auch über Snapshot-Technologien zu duplizieren, ohne dass tatsächlich ein physischer Kopiervorgang anfällt. Dieser Artikel zeigt, warum dies

sinnvoll ist, und gibt einen Überblick über einige der möglichen Varianten. Da es sich hier nur um eine Übersicht handelt, erhebt er keinen Anspruch auf Vollständigkeit oder eine technisch erschöpfende Darstellung. Für tiefergehende Informationen empfiehlt sich daher der Blick in weiterführende Artikel oder in die jeweilige Dokumentation.

Das vollständige Kopieren einer Datenbank mit traditionellen Methoden wird gelegentlich auch als Erzeugen einer „full copy“ oder „fat copy“ bezeichnet. Je nach Größe der Datenbank und Geschwindigkeit des zugrunde liegenden Storage-Systems

kann dieser Vorgang sehr zeitraubend sein. Auch ist das Kopieren einer kompletten Datenbank in der Regel nicht sehr effizient, wenn man bedenkt, dass für die meisten Anwendungsfälle nur sehr wenige Änderungen oder neue Daten in der geklonten Datenbank anfallen werden.

Viel sinnvoller erscheint es daher, einen vollständigen Kopiervorgang zu vermeiden, indem man sich die Eigenheiten von Storage-Snapshots zunutze macht. Man könnte dabei von einer logischen Kopie (auch „thin copy“ oder „thin clone“) sprechen. Vorteile dabei sind vor allem der

```
create pluggable database
testklon-db from prod-db snapshot copy;
```

Listing 1

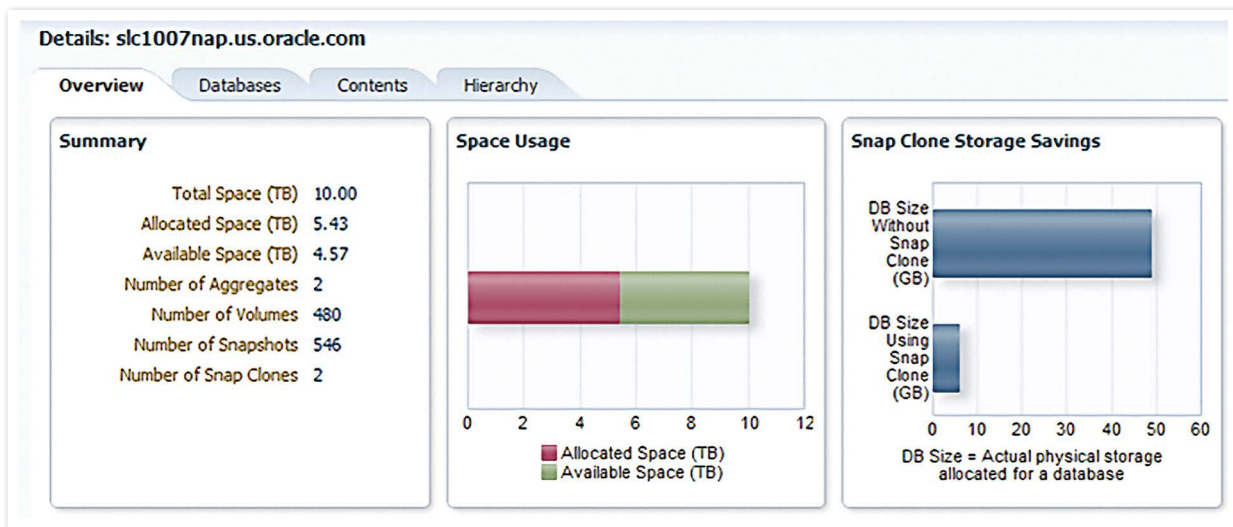


Abbildung 1: Ansicht eines für Snap Clones registrierten Storage-Systems in Enterprise Manager 12c

wesentlich geringere Zeitbedarf und auch eine größere Storage-Effizienz. Die Ersparnis an Zeit und Storage-Platz steigt mit der Größe der jeweiligen Datenbank beträchtlich an. Andersherum ausgedrückt: Je größer eine zu klonende Datenbank ist, desto lohnender ist es, sich mit den Techniken in diesem Artikel zu befassen. Konkret wurden zum Beispiel Zeiteinsparungen im Bereich von 80 bis 90 Prozent sowie Storage-Einsparungen von bis zu 99,9 Prozent beobachtet [1]. Dieser Artikel berücksichtigt folgende Verfahren zum Erstellen von Snapshot-basierten Datenbank-Klonen:

- Manuell/mit Skripten
- Snap-Management-Utility
- Snap Clone in Enterprise Manager Cloud Control
- Snapshot-Copy-Klausel beim Klonen von Pluggable Databases

Am Ende des Artikels sind diese Verfahren in einem tabellarischen Fazit mit ihren jeweiligen Vor- und Nachteilen gegenübergestellt.

Technische Grundlagen und Voraussetzungen

Grundlage aller in diesem Artikel genannten Methoden sind die von modernen Storage- und/oder File-Systemen bereitgestellten „Copy on Write“-Snapshots (CoW). Bei diesem Verfahren werden beim Anlegen eines Snapshot die entsprechenden Daten nicht wirklich kopiert (also physisch verdop-

pelt), sondern es wird, vereinfacht gesagt, lediglich ein neuer Eintrag angelegt, der auf die gleichen Daten verweist. Erst wenn an dem Snapshot Daten geändert werden oder neue hinzukommen, werden die entsprechenden Datenblöcke im Snapshot gesichert beziehungsweise die neuen geschrieben. Da nennenswerte I/O-Operationen erst beim Schreiben oder Ändern von Daten anfallen, erfolgt das Anlegen eines solchen Snapshot extrem schnell.

CoW-Implementationen sind auf unterschiedlichsten Ebenen zu finden: Sie können im Filesystem, im Volume Manager oder auf Storage-Ebene angelegt sein und für ein ganzes Volume, Filesystem oder auch nur einzelne Dateien zur Verfügung stehen. Filesysteme, die entsprechende Features anbieten, sind zum Beispiel ZFS, btrfs, ACFS und OCFS2. Entsprechende Storage-Systeme gibt es zum Beispiel von Oracle oder NetApp.

Hinweis: Je nach konkreter Implementierung ist entweder der Snapshot als solches bereits beschreibbar oder aber es muss auf Basis dessen zunächst ein Klon des Snapshot im Storage beziehungsweise Filesystem erzeugt werden, um darin enthaltene Daten ändern zu können oder neu zu schreiben. Die Nomenklatur ist daher bei der Verwendung der Begriffe „Snapshot“ und „Clone“ nicht immer eindeutig. Der Einfachheit halber wird im Folgenden auf diese Unterscheidung verzichtet und grundsätzlich von beschreibbaren Snapshots/Clones ausgegangen.

Klonen mit selbst erstellten Skripten

Schon seit geraumer Zeit ist es möglich, sich komplett von Hand einen Mechanismus (zum Beispiel ein Skript) zu erstellen, der die (physikalische) Kopie eines konventionellen Datenbank-Klons durch einen Storage-Snapshot der Datenbank ersetzt. Es sind dazu natürlich alle notwendigen Schritte zu berücksichtigen. Konkret handelt es sich im Wesentlichen um:

- Herunterfahren oder zumindest „Anhalten“ der Datenbank (mittels „...begin backup“)
- Erzeugen des Snapshot durch Aufruf entsprechender Storage-Befehle
- Anlegen beziehungsweise Kopieren der notwendigen Datenbank-Metadaten und -Konfiguration
- Konfigurations-Anpassung (wie Umbenennungen zur Vermeidung von Namenskollisionen)

Da die Einrichtung und Pflege eines solchen Verfahrens relativ mühsam und zeitraubend sein kann, wird es an dieser Stelle auch nicht näher betrachtet. Allen Lesern, die dennoch an einer individuellen, skriptbasierten Lösung interessiert sind, sei an dieser Stelle eine Variante empfohlen, die die mit Oracle 11.2.0.2 als Bestandteil von Direct NFS (dNFS) eingeführte „clonedb“-Funktionalität nutzt. Hierbei implementiert der dNFS-Stack der Datenbank selbst einen CoW-Mechanismus auf Basis einer

Verfahren	Unterstützte Storage- & Filesysteme	Unterstützte Datenbank-Versionen	Benötigtes technisches Know-how	Grafische Oberfläche	Zusätzliche Software-Installation erforderlich	Produkt/ unterstützte Lösung
Eigene Skripte (z.B. mittels dNFS-Clone)	Nahezu beliebig (btrfs, OCFS2, ZFS, ...)	ab 11.2.0.2	hoch	nein	nein	nein
SMU	Nur ZFSSA	(10g*), 11g	gering**	ja	ja	ja
Snap Clone	ZFSSA, NetApp, generischer ZFS-Storage	(10g*), 11g, 12c	gering**	ja	ja	ja
Snapshot-Copy für PDBs	ZFSSA, NetApp, ACFS	Nur 12c mit Multitenant Option	mittel	Je nach Aufruf/ Nutzung von SQL	nein	ja

Tabelle 1

* Oracle 10g würde prinzipiell/technisch auch unterstützt, ist aber nicht mehr im aktiven Support

** Initiale Einrichtung kann Hilfe von Storage-Administrator erfordern

zuvor entweder physisch kopierten oder als Snapshot (zum Beispiel auf btrfs) erzeugten Datenbank-Kopie. Beschrieben ist dies unter anderem in einigen Blog-Postings und Artikeln, so zum Beispiel in der DBA Community [2] sowie in einem auf der letztjährigen DOAG-Konferenz gehaltenen Vortrag [3].

Snap-Management-Utility für Oracle ZS3/ZFSSA

Beim sogenannten „Snap Management Utility“ (SMU) handelt es sich um ein von Oracle angebotenes Software-Werkzeug, das die Erstellung von Snapshot-basierten Datenbank-Klonen auf Storage-Systemen der ZS3-Reihe (auch bekannt als „ZFS Storage Appliance“) bietet. Es kapselt die Möglichkeiten dieser Systeme zum Erstellen von Snapshot-basierten Datenbank-Klonen in einer einfach zu bedienenden grafischen Management-Konsole. Ziel ist es, dass DBAs dadurch auch ohne genaue Kenntnisse der zugrunde liegenden Storage-Mechanismen schnell und einfach DB-Klone erstellen können. Nähere Informationen zum SMU finden sich im Artikel in dieser Ausgabe auf Seite 18 sowie unter [4] und [5].

„Snap Clone“-Feature in Enterprise Manager 12c Cloud Control

Enterprise Manager 12c Cloud Control bietet im Rahmen des „Cloud Management Packs for Oracle Databases“ die Möglichkeit, neue, aber bereits mit Inhalten versehene Datenbanken durch einen User im Self-Service anzufordern. Die Datenbanken können hierbei nicht nur konventionell

erzeugt werden (also durch physisch kopierte Datenbank-Dateien), sondern auch auf Basis von Storage-Snapshots. Dieses Feature wird als „Snap Clone“ bezeichnet. Konkret unterstützt werden dafür zurzeit:

- Systeme der ZS3-Serie
- NetApp Filer
- Generische Storage-Systeme (SAN/NAS), die als ZFS-Dateisystem von einem Oracle Solaris Server (ab Solaris 11.1) bereitgestellt sind

Als Grundvoraussetzung für diese Funktionalität muss der jeweilige Storage einmalig in Enterprise Manager registriert sein. Dazu ist zuvor sowohl das Plug-in für das „Storage Management Framework“ sowie das Plug-in für das jeweilige Speichersystem zu installieren. Damit dies auch funktioniert, sind natürlich auf dem Storage-System noch ein paar Handgriffe durch dessen Administrator erforderlich, um die entsprechenden Storage-API-Funktionen auch einzurichten beziehungsweise freizuschalten. Die genauen Schritte sind in der Enterprise-Manager-Dokumentation beschrieben [6].

Sobald die genannten Voraussetzungen erfüllt sind, kann man im „Database as a Service“-Bereich von Cloud Control an zwei Stellen von Snap Clones profitieren. Zum einen ist es möglich, auf Basis von Storage-Snapshots sogenannte „Database Profiles“ anzulegen, die eine bereits existierende Datenbank repräsentieren, um diese später sehr schnell als Klon bereitzustellen. Wenn ein Cloud-Administrator also bei der

Definition eines „Service Templates“ für eine Datenbank ein solches Snapshot-basiertes Profil ausgewählt hat, erfolgt später für einen Nutzer, der das entsprechende Template im Self-Service-Portal auswählt, kein Kopiervorgang mehr. Stattdessen wird die Datenbank als „Snap Clone“ provisioniert, was unabhängig von der Größe des Originals immer gleich schnell erfolgt. Änderungsdaten beziehungsweise neue Daten in dieser Datenbank sind für den Nutzer auf eine zuvor vom Cloud-Administrator definierte Größe limitiert („writeable space“) [7]. Die dadurch erzielten Storage-Einsparungen sind für den Administrator jederzeit in Oracle Enterprise Manager ersichtlich, wie *Abbildung 1* zeigt.

Zum anderen kann ein Cloud-Nutzer im Self-Service-Portal Backups als logische Kopien seiner von ihm provisionierten Datenbank anlegen, etwa um nach einem Testvorgang später wieder auf einen definierten Zustand aufsetzen zu können. Normalerweise werden diese Backups mithilfe klassischer Methoden durch archivierte Redologs und ein RMAN „point in time restore“ zum späteren Zeitpunkt durchgeführt. Alternativ kann der Nutzer aber, sofern es der Administrator zuvor eingestellt hat, auch hier die „Snap Clone“-Funktionalität nutzen. Ein Zurücksetzen der Datenbank erfolgt dann nicht mittels RMAN, sondern durch Wiederherstellen eines früheren Snapshot, was bei größeren Datenbanken natürlich wesentlich schneller geht. Es sei an dieser Stelle allerdings die Anmerkung angeführt, dass ein Snapshot kein richtiges Backup,

sondern lediglich ein logisches Backup ist. Eine korrupte Datenbank kann mit einem solchen Backup nicht wiederhergestellt werden, da korrupte Blöcke dann auch im Snapshot korrupt sind.

„Snapshot Copies“ mit PDB-Cloning bei 12c-Multitenant

Ein wesentlicher Bestandteil der mit der Datenbank 12c neu eingeführten Multitenant-Architektur ist die Tatsache, dass sich eine Pluggable-Datenbank (PDB) nicht nur leer, sondern auch als Klon einer bestehenden PDB erzeugen lässt. Diese Cloning-Möglichkeit ist bereits direkt in den entsprechenden SQL-Befehl eingebaut, sodass im einfachsten Fall ein „create pluggable database <name-des-klons> from <original-pdb>“ bereits zum Anlegen einer geklonten PDB ausreicht. Hierbei wird zunächst jedoch nicht auf Snapshot-Mechanismen zurückgegriffen, sondern ein normaler Kopiervorgang der zugrunde liegenden Dateien durchgeführt.

Durch ein einfaches Anhängen der Klausel „Snapshot Copy“ lässt sich dieses Verhalten jedoch ändern, um völlig transparent auch von einer vorhandenen „Storage-Intelligenz“ zu profitieren. Ein kompletter Aufruf könnte also wie in *Listing 1* aussehen.

Konkret unterstützte Storage-Systeme, die diese „Snapshot Copy“ anbieten sind:

1. Oracle ZS3/ZFSSA
2. NetApp Filer
3. Oracle ACFS (ASM Cluster File System, auch als Cloud-Filesystem bekannt)

Voraussetzung für die Nutzung der ersten beiden Storage-Systeme ist das einmalige Hinterlegen der jeweiligen Storage-Credentials im Wallet der Container-Datenbank, da sonst der DBA ohne Zutun eines Storage-Administrators die Snapshot-Copys nicht anlegen könnte. Wie auch bei normalem PDB-Cloning muss die Quell-PDB in den „Read only“-Modus versetzt sein, bevor man den Klon tatsächlich anlegen kann. Aus naheliegenden Gründen kann eine PDB außerdem nicht mehr gelöscht werden, solange Klone von ihr existieren. Weitere Informationen dazu stehen zum Beispiel im Abschnitt „Thin Provisioning and Instantaneous Cloning: The Oracle ZFS Storage Appliance Advantage“ auf OTN [8].

Fazit

Wie dieser Artikel zeigt, haben alle hier vorgestellten Verfahren zum Datenbank-Cloning ihre entsprechenden Eigenarten und Voraussetzungen. Einige erfordern zum Beispiel den Einsatz bestimmter Datenbank-Versionen oder -Optionen, andere funktionieren vielleicht gerade mit dem eigenen Lieblings-Storage nicht. Welche der vorgestellten Varianten am besten passt, sollte daher je nach vorhandenem Know-how sowie der jeweils vorhandenen IT-Infrastruktur entschieden werden. *Tabelle 1* liefert dabei eine kleine Entscheidungshilfe.

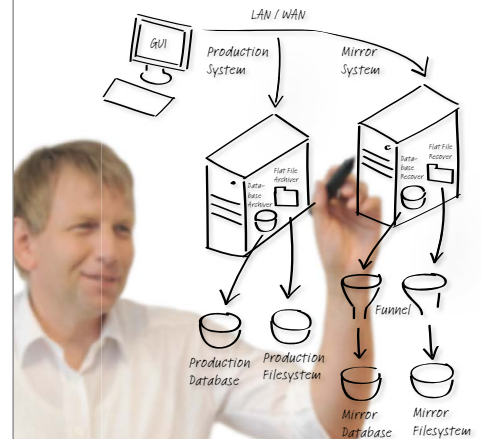
Weiterführende Links

- [1] <http://www.oracle.com/technetwork/articles/servers-storage-admin/multitenant-on-sparc-solaris-2016889.html>
- [2] Sebastian Solbach, „Direct NFS Clone – Klonen der Oracle Datenbank zum Testen“: https://apex.oracle.com/pls/apex/GERMAN_COMMUNITIES.SHOW_TIPP?P_ID=1121
- [3] Robert Marz, „Da fliegt die Kuh – Rasante Datenbank-Klone durch cow (copy on write)“, abrufbar für Teilnehmer der DOAG 2013 Konferenz im Mitgliederbereich der DOAG-Website
- [4] <http://www.oracle.com/us/products/servers-storage/storage/nas/snap/overview/index.html>
- [5] <http://www.oracle.com/us/products/servers-storage/storage/nas/smu-bus-wp-final-1857065.pdf>
- [6] Oracle Enterprise Manager Cloud Administration Guide 12c Release 3, „Registering and Managing Storage Servers“: http://docs.oracle.com/cd/E24628_01/doc.121/e28814/cloud_db_setup.htm#BABGDDBB
- [7] Oracle Enterprise Manager Cloud Administration Guide 12c Release 3, „DBaaS Using Snap Clone Based Database Provisioning Profile“: http://docs.oracle.com/cd/E24628_01/doc.121/e28814/cloud_db_portal.htm#CHDBCIIIG
- [8] „Oracle Multitenant on SPARC Servers and Oracle Solaris“: <http://www.oracle.com/technetwork/articles/servers-storage-admin/multitenant-on-sparc-solaris-2016889.html>



Manuel Hoßfeld
manuel.hossfeld@oracle.com

Libelle BusinessShadow®



Unabhängig bezüglich

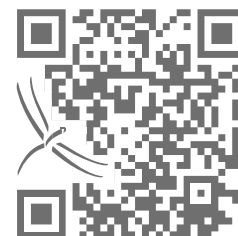
- Fehlerursache
- Entfernung
- Hardware / Architektur
- Komplexer Systeme

Schnelle Arbeitsaufnahme

- Mit konsistenten Daten
- Auf Knopfdruck
- Automatisiert
- ...

Hans-Joachim Krüger
Chief Technology Officer
Libelle AG

Erfahren Sie mehr:
www.Libelle.com/business



ORACLE Gold Partner



Libelle

Libelle AG

Gewerbestr. 42 • 70565 Stuttgart, Germany
T +49 711 / 78335-0 • F +49 711 / 78335-148
www.Libelle.com • sales@libelle.com