Database Virtualization and Consolidation Technologies

Kyle Hailey
Average customer makes 12 copies of production
- Charles Garry, Oracle

Database Virtualization consolidates copies of production
Database Virtualization

• Comes of age
  – CloneDB: 3 talks @ OOW 2012
    • Clone Online in Seconds with CloneDB (EMC)
    • CloneDB with the Latest Generation of Database (Oracle)
    • Efficient Database Cloning with Clonedb (Cern)
  – Oracle 12c: new feature
  – Companies:
    • Delphix
    • NetApp
    • VMware

• What is it?
  – database virtualization is for data tier
    as VMware is for compute tier
  *** not putting Oracle on a VM host
CERN - European Organization for Nuclear Research
- 145 TB database
- 75 TB growth each year
- 10s of developers want copies.
Copies – time consuming and space consuming

• Time consuming
  – Time to make copies, days to weeks
    • RMAN backup, archive logs, copy day over, recover
  – Meetings, days to weeks
    • System, Storage, Database, Network Admins, manager coordination

• Space consuming
  – 100 devs x 1TB production = 100TB
    • This is 100x actual unique data
    • Unique data is
      – 10 TB original
      – 2TB of changed data
      ➢ = 12TB total unique data
Copy workarounds

Sub set copy share

Developers

QA and UAT

Production

First Copy

Shared stale

Reports
Copy work around problems

• Share copies -> slow projects down
  – Shared copies slow down development, days to weeks to check in changes
  – Long delays -> stale copies -> wrong results
  – Hard to get a new copy if everyone is sharing current copy

• Subset copies -> misleading and/or wrong
  – Incomplete results
  – Performance results may be wrong
solution:

thin provision cloning, ie "virtualized databases"

Instead of full copies of same data

One Read Only Copy plus thin layer of changes per clone

Clone manages modified data

Clone, Share, Consolidate
Virtualization: Consolidation and agility

Clones: fast to create, small footprint, can create from any point in time
Technologies:

1. CloneDB (Oracle)
2. ZFS Storage Appliance (Oracle)
3. Delphix
4. Data Director (Vmware)
5. EMC
6. NetApp
7. Oracle 12c Snap Manager Utility (SMU)
Virtualization: Advantages

• Space
  – Clones sharing a single snapshot
    • 100 copies of 1TB goes from 100 down to 1 TB (0.3TB with compression)

• Speed
  – Eliminate Coordination
    • System, Storage, Database, Network Admins + manager coordination
  – Eliminate copying
  – Creation time of clone quick

-> Agility
Barriers to Entry

• Hardware
  – specialized equipment required
  – NetApp, EMC, ZFS Storage Appliance

• Build it yourself
  – expert knowledge
  – brittle scripting

• Self service Provisioning of VDBs
  – Empowering none DBAs to create copies
  – Create, clone, branch, destroy, quotas, access control on source and target
You Should be cloning now

If you have any of:

- Oracle 11.2.0.2+
- Oracle ZFS Storage Appliance
- NetApp

You should be cloning. Cloning gives you

- Storage savings
- More importantly time savings
  -> Agility

How many copies are of database are made?
What size are these databases?
How often are the copies made?
What do the technologies offer?

1. Snapshot
   - All (some more limited than others)

2. Clone
   - All (some more limited than others)

3. Roll Snapshot forward
   - NetApp, Delphix, ZFS

4. Provision
   - Oracle12c, Delphix

5. Automate
   - Delphix
Types of solution – (part 1)

• Hardware Vendor verses Software
  – Hardware lock in: EMC, NetAPP, Oracle ZFS Storage Appliance
  – Software: CloneDB, Delphix, Data Director

• Database Specific versus General purpose Copies
  – Oracle Specific: CloneDB
  – General Purpose: EMC, NetApp, Oracle ZFS Appliance, Data Director
  – Multi Database Specific: Delphix*

*Oracle, SQL Server, User Data, other DBs coming
Types of solution – (part II)

• Golden Copy
  – Required: EMC, DataDirector, CloneDB
  – Not Required: Delphix, Oracle ZFS Appliance, NetApp (snaps of snaps)

• Performance Issues
  – Data Director
  – CloneDB
Types of solution – (part III) : Automation

- Source database changes
  - incremental backups, redo collection, retention windows, expose file systems
- Create databases from clones
  - assigning SID, Parameters, file structure, recovery, Security
- Cloud ready
  - Hardware agnostic, Multi database support Oracle, SQL Server, Sybase, DB2, PostGres, MySQL
  - Masking data
  - Load Balancing, Provision databases on hardware with available resources
CloneDB

1. RMAN backup (local or NFS)
2. Create an NFS mount (somewhere)
3. Setup dNFS and 11.2.0.2+ (on target machine)
4. Clonedb.pl initSOURCE.ora output.sql
5. sqlplus / as sysdba @output.sql

Tim Hall

www.oracle-base.com/articles/11g/clonedb-11gr2.php
CloneDB

Setup dNFS and 11.2.0.2+
- libnfsodm11.so
- /etc/oranfstab

Clonedb.pl initSOURCE.ora output.sql
- export MASTER_COPY_DIR="/backuplocal"  # backup location
- export CLONE_FILE_CREATE_DEST="/clone"  # requires NFS MOUNT
- export CLONEDB_NAME="clone"  # ORACLE_SID="clone"

sqlplus / as sysdba @output.sql
- startup nomount PFILE=/clone/initclone.ora
- Create control file with backup location
- dbms_dnfs.clonedb_renamefile(
  '/backup/sysaux01.dbf', '/clone/ora_data_clone0.dbf');
- alter database open resetlogs;

Tim Hall
www.oracle-base.com/articles/11g/clonedb-11gr2.php
Clone DB: requires dNFS and 11.2.0.2+

1. physical

Three machines
1. Physical
2. NFS Server
3. Target

Problem: No Versioning

830264 /backup/sysaux01.dbf
727764 /backup/system01.dbf
425388 /backup/undotbs01.dbf
760 /clone/ora_data_clone0.dbf
188 /clone/ora_data_clone1.dbf
480 /clone/ora_data_clone2.dbf

du -sk
Clone DB: everything could be on NFS

Target A
- Clone 1
- Clone 2
- Clone 3

NFS Server
- Clone 1
- Clone 2
- Clone 3

Read only
- Clone 4
- Clone 5
- Clone 6

Target B
- Clone 4
- Clone 5
- Clone 6

physical

RMAN
Clone DB: refresh: either destroy or duplicate

physical

RMAN

Target A
Clone 1
Clone 2
Clone 3

NFS Server
Clone 1
Clone 2
Clone 3
Clone 4
Clone 5
Clone 6
Read only
Level 0 + 1

Target B
Clone 4
Clone 5
Clone 6
1. ZFS Appliance
   - Create backup project **db_master**
     • With 4 file systems: datafile, redo, archive, alerts
   - Create project for **db_clone** (with same 4 filesystems)

2. Source Database
   - NFS Mount Backup locations from ZFS Appliance
   - Backup with RMAN as copy, archive logs as well

3. ZFS Appliance
   - Login to Appliance shell, Snapshot backup location
     • Select **db_master**
     • Snapshots snapshot snap_0
     • Then each filesystem on db_master clone it onto **db_clone**

4. Target Host
   - Mount **db_clone** directories over NFS from ZFS Appliance
   - Startup and recover clone
Oracle ZFS Appliance

1. physical

ZFS Storage Appliance

Snapshot

Clone 1

RMAN Copy to NFS mount

RMAN copy

NFS

Target A

Clone 1

ZFS snapshot
instantaneous
read only

ZFS Clone
instantaneous
read write
Oracle ZFS Appliance: RMAN incremental

- Production
- ZFS Storage Appliance
- Full Backup
- Incremental Backups
- RMAN Incremental Backups
- Target A
  - Clone 1
  - Clone 2
  - Clone 3
- Target B
  - Clone 1
  - Clone 2
  - Clone 3
  - Clone 4
- Target C
  - Clone 3
  - Clone 4
ZFS

- Prehistory: 1 disk = 1 filesystem
- ~1990: volume managers: N disks : 1 FS
- 2001-2005: ZFS development
- 2005: ZFS ships, code open-sourced
- 2008: ZFS storage appliance ships
  - ZFS enables several ZFS-based startups including Delphix, Nexenta, Joyent
- 2010: ZFS development moves to Illumos
  - headed by Delphix
FS/Volume Model vs. Pooled Storage

Traditional Volumes
- Abstraction: virtual disk
- Partition/volume for each FS
- Grow/shrink by hand
- Each FS has limited bandwidth
- Storage is fragmented, stranded

ZFS Pooled Storage
- Many filesystems in one pool
- No partitions to manage
- Grow automatically
- All bandwidth always available
- All storage in the pool is shared
Always consistent on disk (COW)

1. Initial block tree

Meta Data

Uber Block

File Systems

Dnodes

File

Data Blocks
Always consistent on disk (COW)

1. Initial block tree

2. COW some blocks

3. COW indirect blocks

4. Rewrite uberblock (atomic)
Bonus: Snapshots

Snapshot root (file system) -> Live root

[Diagram showing a tree structure with nodes labeled as Snapshot root and Live root, with various green and purple boxes representing different levels of the file system hierarchy.]
Bonus: Constant-Time Snapshots

- Younger snapshots than blocks => keep
- No younger snapshots => free

Sync writes are written immediately out to Intent log
Data and Metadata is batch written out later
ZFS Data Relationships

• Snapshot is a read-only point-in-time copy of a filesystem
  - Instantaneous
  - Unlimited
  - No additional space

• Clone is a writable copy of a snapshot
  - Instantaneous
  - unlimited
  - No additional space

• Send / receive: replication
  - Can send a full snapshot
  - Can send incremental changes between snapshots
  - Incremental send/receive quickly locates modified blocks
ZIL (ZFS Intent Log) Overview

• ZIL is per filesystem
• Logs filesystem modifications
• Log can used to replay filesystem changes
  • In the event of power failure / panic, the log records are replayed
• Log records are stored in memory until:
  o Sync write, ie fsync() or O_DSYNC
  o Transaction group commits
ZFS at Delphix

- **Compression**
  - typically ~2-4x

- **Block sharing**
  - Via clones, Faster, cheaper than Deduplication which is too slow with overhead

- **Link Source DB**
  - create new filesystems for datafile, archive, etc.
  - set recordsize of datafile FS to match DB

- **Snapshot Source**
  - take ZFS snapshot of datafile fs
  - retain relevant log files in archive fs

- **Clone Provision VDB**
  - create clone of Source's datafile snapshot
  - share the dSource's blocks; no additional space used
  - new data takes space
Delphix

RMAN backup sets
- Allows control over send
- Unused blocks not sent

Delphix
- Rebuilds the datafiles
- Rebuilds unused blocks
- Compresses datafiles
- Highly compressed zero regions, 2-4x compression
Delphix

Production

RMAN

Full Backup

Free-able

Incremental Backups

Redo

Delphix Appliance

Snapshot

Clone 1

Clone 2

Target A

Clone 1

Target B

Clone 2
Performance Improvement of Database Virtualization

Graph showing comparison between physical and virtual database configurations:
- Physical: SAN, Cache, Target1 Database
- Virtual: SAN, Cache, IBM 3690, Delphix, Target1 Database

Bar chart comparing TPM (Transactions Per Minute) across different conditions:
- Physical
- v1 Cold
- v1 Warm

Users: 20
Multiple Virtual sharing cache

Physical
- SAN
- Target1 database
- Target2 database

Virtual
- SAN
- Delphix
- Target1 database
- Target2 database

Graph:
- TPM vs. Users
- Physical
- physical1 concur
- physical2 concur
- v1
- v1 concur
- v2 concur

Users
Multiple Virtual sharing cache

Seconds

![Bar chart showing performance metrics for different queries and cache states.](chart.png)
NetApp

Production Database

NetApp Filer

Database Luns

snapshot

File system level

NetApp Filer

snapshot

clones

Target A

Clone 1

Target B

Clone 2

Target C

Clone 3

Clone 4
NetApp

Production Database

NetApp Filer

Database Luns

snapshot

NetApp Filer

snapshot

NetApp Filer

snapshot

Target A

Clone 1

Target B

Clone 2
NetApp

Physical Database

NetApp Filer

Database Luns

snapshot

NetApp Filer

snapshot

NetApp Filer

Target A

Clone 1

Target B

Clone 2

snapshot

snapshot

snapshot
NetApp Limits

Limit of 255 snapshots
snaps are limited to the same aggregate (storage pool)
Aggregates have size limits depending on controller

<table>
<thead>
<tr>
<th>Controller</th>
<th>Size Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>32 bit controllers</td>
<td>16TB</td>
</tr>
<tr>
<td>FAS3140/FAS3040/FAS3050</td>
<td>40TB</td>
</tr>
<tr>
<td>FAS3160/FAS3070</td>
<td>50TB</td>
</tr>
<tr>
<td>FAS6040/FAS3170</td>
<td>70TB</td>
</tr>
<tr>
<td>FAS6080</td>
<td>100TB</td>
</tr>
</tbody>
</table>

All sources have to be in the same aggregate to be snapshot together.
EMC

- Point of view: DR, backup and testing off of a full copy
  - Create BCV, a full copy
  - Promote BCV to make accessible
  - Take snaps of BCV (limit 32?)
  - Zone and mask LUN to target host
  - Full copy of disk, now recover (may have to rename the LUNs)

- “Golden Copy”
  - Journal based filesystem (Not a pointer based like NetApp and ZFS)
  - EMC uses a save area, the amount of area for changes to the snapshot
  - No time flow
  - Initial snapshot has to stay
  - To get rid of “golden copy” have to recreated it with the new changes

- Snapshots
  - 8 snapshots and then have to rebuild
  - Can’t take a snap of a snap on low end
  - Can only take one level snap of a snap on high end
Data Director : Linked Clones (Vmware)

- **Golden Copy issue**
  - original copy has to always exist

- **x86 host databases only**
  - Linux
  - OpenSolaris

- **Performance issues**
  - “Having several linked clones can affect the performance of the source database and the performance of the linked clones.”
  - “If you are focused on performance, you should prefer a full clone over a linked clone.”
  - Performance worse with more snapshots
  - Can only take 16 snapshots
  - Performance worse with more concurrent users
Oracle 12c

- Oracle Snap Manager Utility for ZFS Appliance
- Pay for option
- Requires source database hosted on ZFS appliance
- Principally a manual GUI
  - utility to snapshot source databases and provision virtual databases
- No concept of time flow
  - Virtual databases have to be provisioned of snapshots.
Conclusion: Enterprise Solutions

- **EMC Timefinder, VMware Data Director**
  - offer limited ability to benefit from cloning
- **Clonedb ***
  - fast easy way to create many clones of the same copy
  - limited to 11.2.0.2 and systems with sparse file system capability
  - suffers the golden image problem and performance
- **NetApp Flexclone, Snap Manager for Oracle**
  - offers a rolling solution
  - limited database awareness
  - file system clones
  - limited snapshots
  - Vendor lock-in
- **Oracle ZFS Appliance**
  - Vendor Lock-in
- **Delphix**
  - Agility: Automation, unlimited snapshots, clones, multi-database
## Matrix of features

<table>
<thead>
<tr>
<th></th>
<th>CloneDB</th>
<th>ZFS Appliance</th>
<th>Delphix</th>
<th>Data Director</th>
<th>NetApp</th>
<th>EMC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time Flow</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Hardware Agnostic</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Snapshots</td>
<td>No</td>
<td>Unlimited</td>
<td>Unlimited</td>
<td>31</td>
<td>255</td>
<td>16 (96 read only)</td>
</tr>
<tr>
<td>Snapshots of snapshots</td>
<td>No</td>
<td>Unlimited</td>
<td>Unlimited</td>
<td>30</td>
<td>255</td>
<td>1</td>
</tr>
<tr>
<td>Automated Snapshots</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Automated Provisioning</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Any DB host O/S</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No x86 only</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Max DB size</td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>~200G</td>
<td>16-100TB</td>
<td>?</td>
</tr>
</tbody>
</table>
Appendix

- CloneDB
- ZFS
  - http://hub.opensolaris.org/bin/download/Community+Group+zfs/docs/zfslast.pdf
- ZFS Appliance
- Data Director
- EMC
- NetApp
  - RAC provision example http://blog.fli matech.com/2008/02/07/how-to-create-a-netapp-flexclone-rac-database/
• END