Database Virtualization

and

Instant Cloning

http://dboptimizer.com

DELPHIX®

Its an exciting **time of growth** in the industry.
The growth of the industry is **accelerating**.
Its almost like the next **dot com** but
dot com with a solid foundation
garnter report
For companies to **keep up** with this acceleration requires **innovation**
--
Our company **creates** software
That **provisions** oracle database clones
minutes
no overhead
Achieved by sharing **duplicate**
--
How does this technology affect the industry?
I like to ask companies
- sufficient storage
- enough DBAs
- is project output as high as they want and on time

Delphix
reduces storage excesses

alleviates DBA of repetitive work of
copying file
allowing them to focus on interesting work
and **innovation**

put a jet pack on database development
Eliminate the development **bottleneck** of
creating development evironments
Allowing developers to write **more code**
faster and of better **quality**

put a jet pack on database development
Eliminate the development **bottleneck** of creating development evironments
Allowing developers to write **more code faster** and of better **quality**

Delphix replaces burdensome storage requirements of database copies with fast free provisioning through automated agile software.

Average customer makes 12 copies of production

- Charles Garry
Database Product Manager Oracle

Expect this to double by the time 12c is fully adopted

Two Parts

1. Cloning Technology
2. Development Acceleration

Cloning Technology

Physical **Thin Provision** **Virtual**

1. Physical Clones

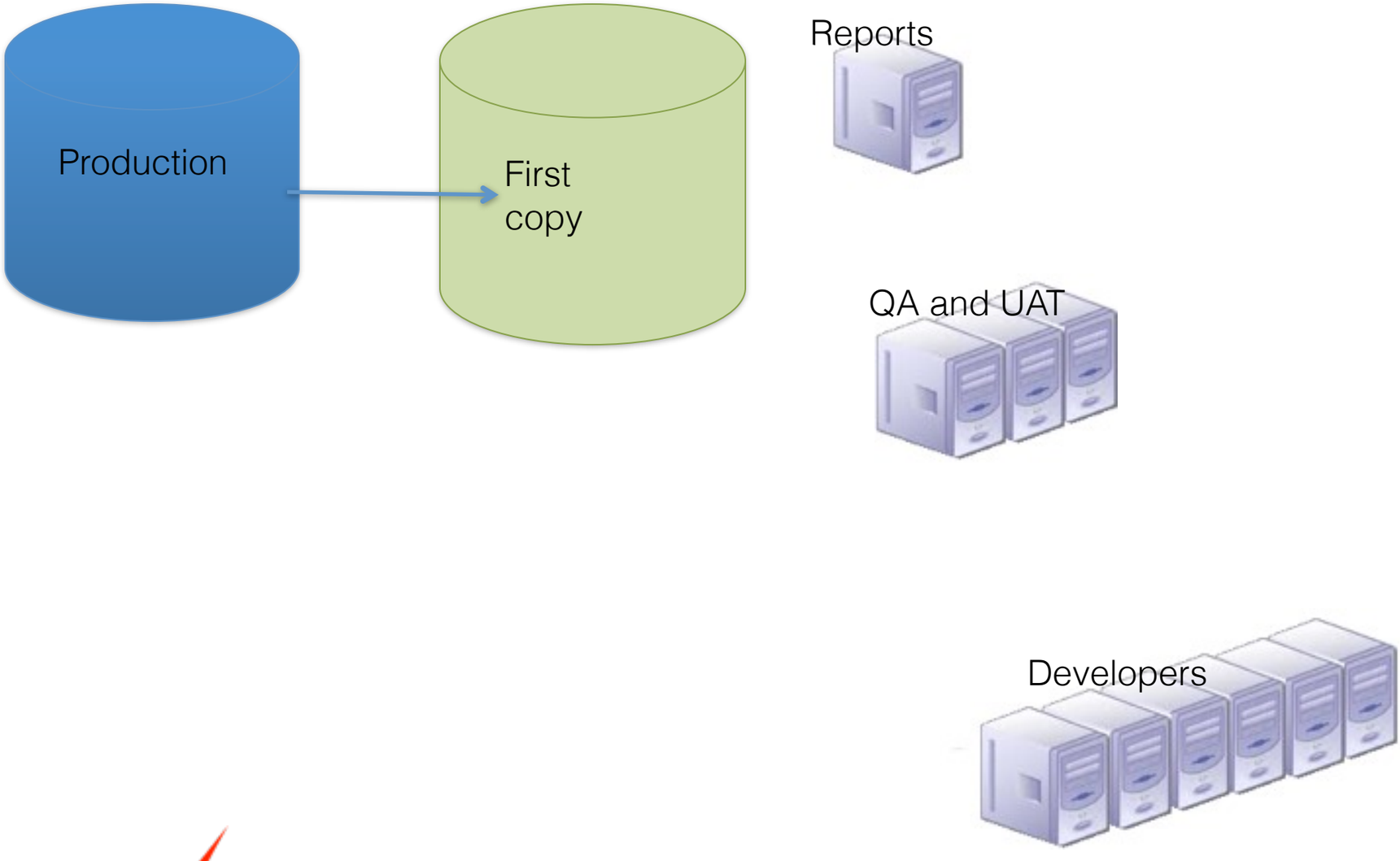
Business wants data now

If the can't get access

Then the process is broken

Business doesn't understand DBAs and their work

Problem

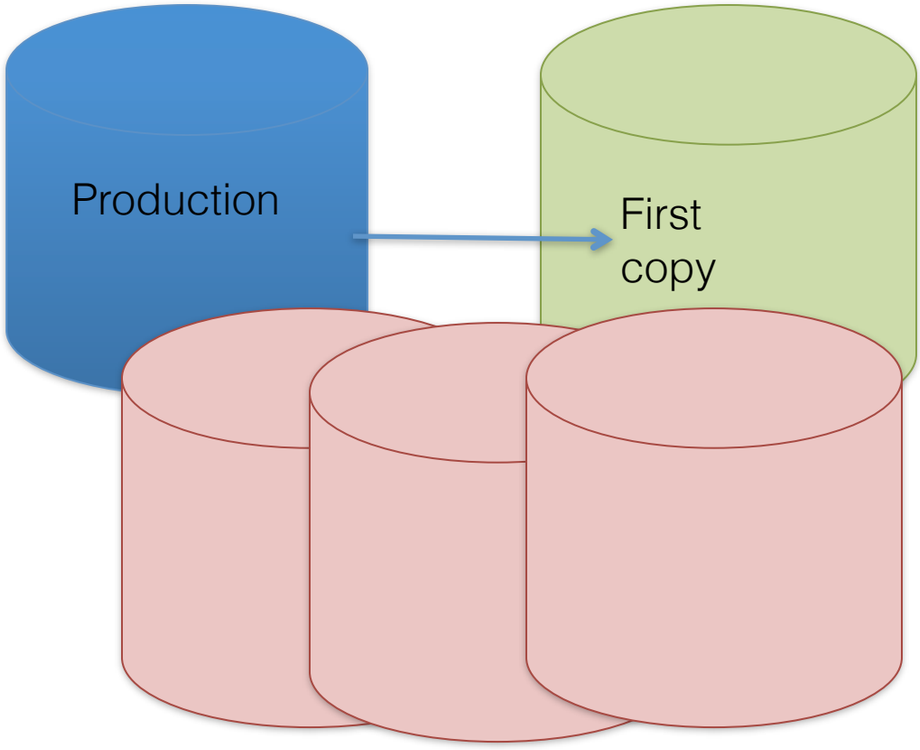


DELPHIX

•Reporting
•QA and UAT
•Developers

Groups have different needs and usage patters so need separate copies.
Some sharing can be accomplished between users of the same type, but this slows down productivity.

Problem

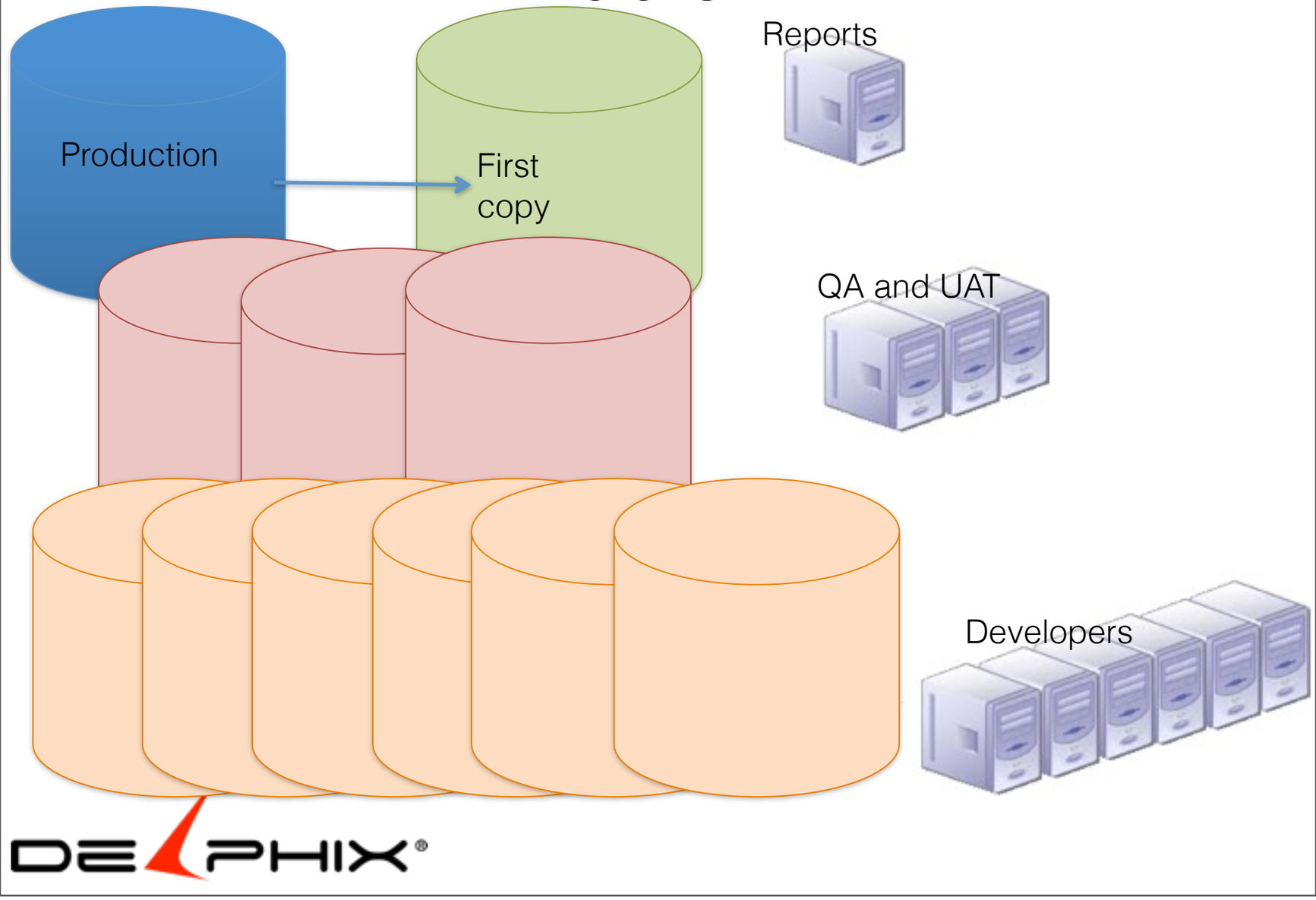


DELPHIX®

•Reporting
•QA and UAT
•Developers

Groups have different needs and usage patters so need separate copies.
Some sharing can be accomplished between users of the same type, but this slows down productivity.

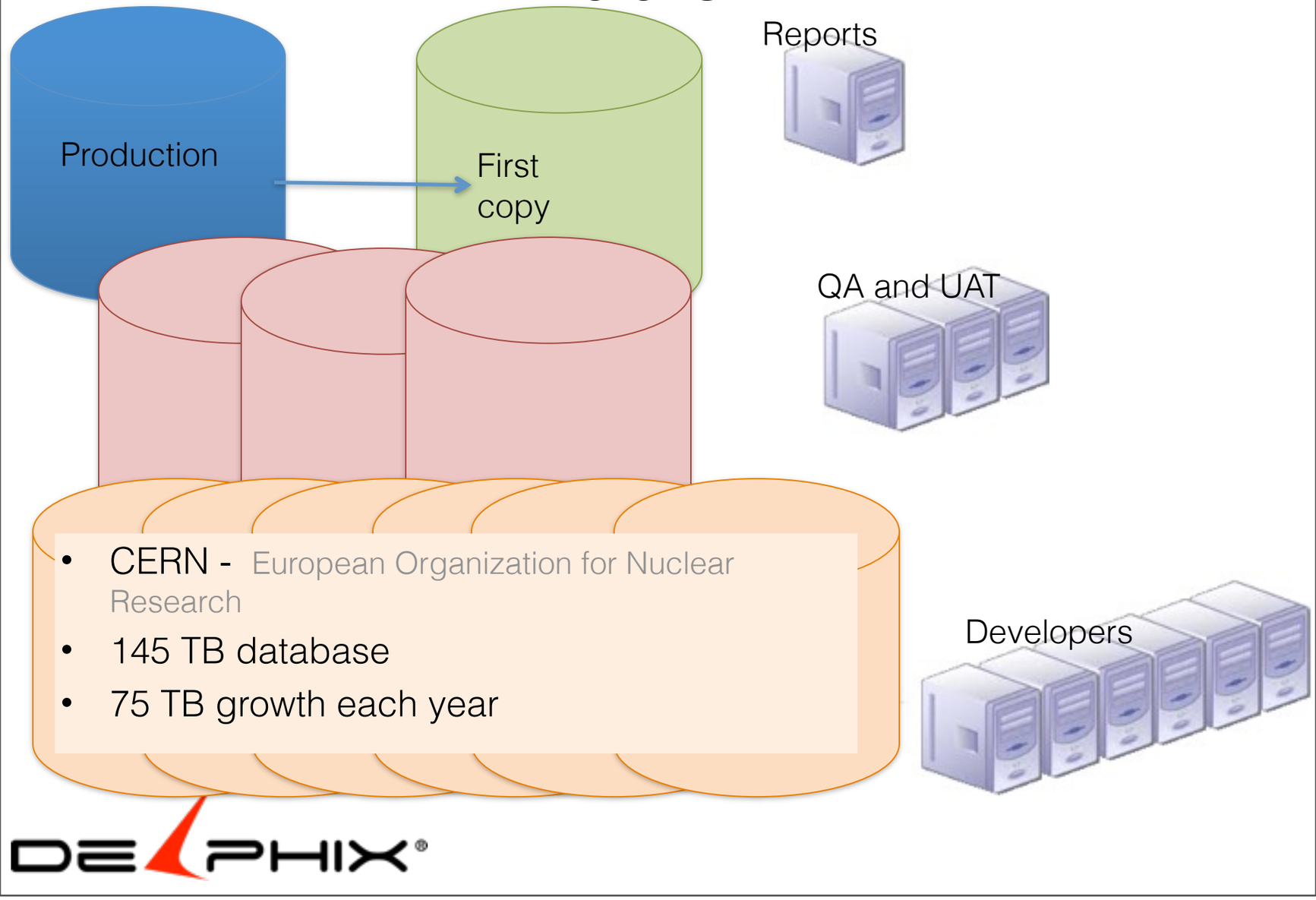
Problem



•Reporting
•QA and UAT
•Developers

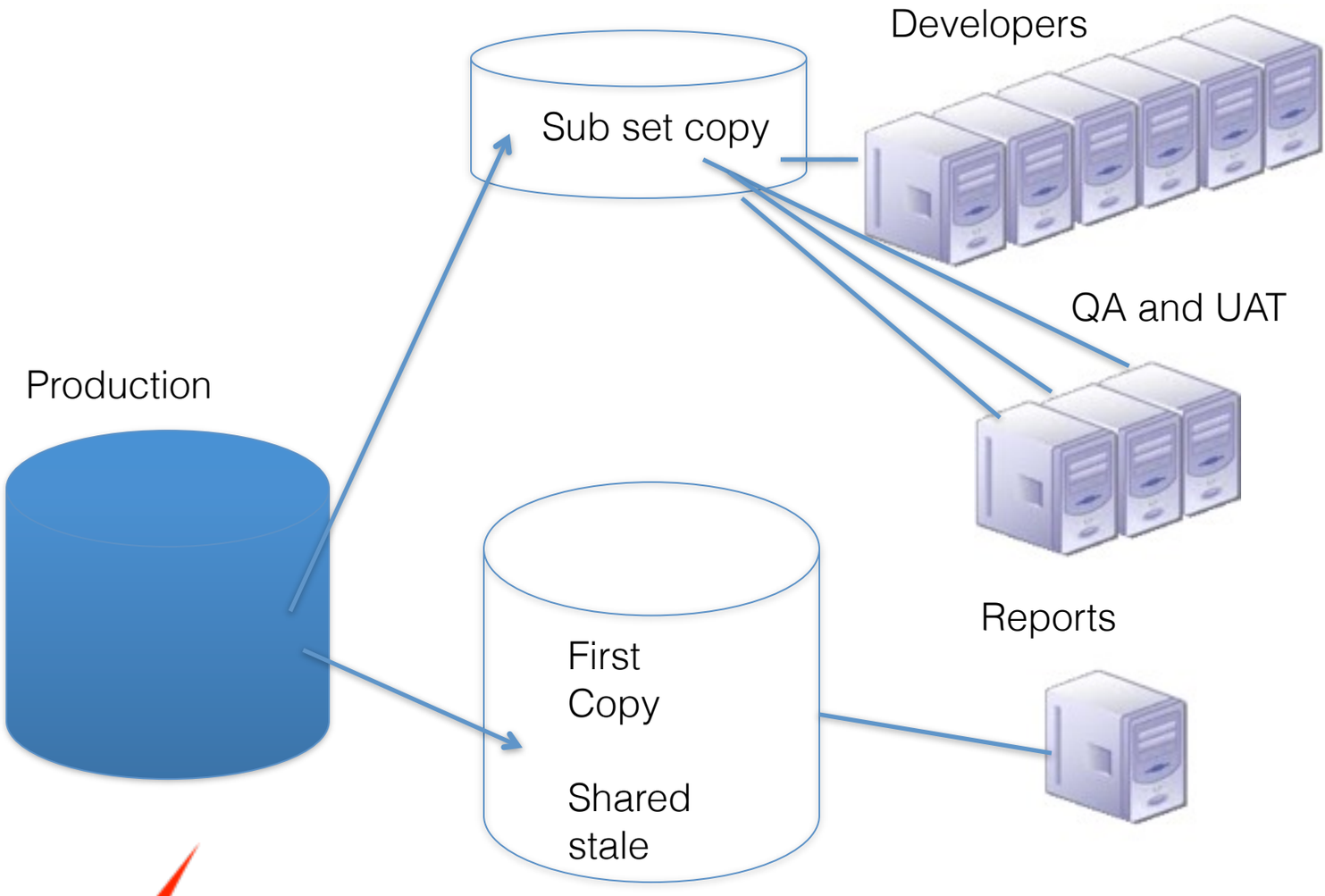
Groups have different needs and usage patters so need separate copies.
Some sharing can be accomplished between users of the same type, but this slows down productivity.

Problem



•Reporting
•QA and UAT
•Developers
Groups have different needs and usage patters so need separate copies.
Some sharing can be accomplished between users of the same type, but this slows down productivity.

workarounds



DELPHIX®

Making copies of production databases for
•Reporting
•QA and UAT
•Developers
Groups have different needs and usage patters so need separate copies.
Some sharing can be accomplished between users of the same type, but this slows down productivity.

database

= heavy

DELPHIX®

Databases are **large**
Moving data around is **hard** work.
Moving them takes **time, resources, equipment** as well as know how.

The business doesn't know or **understand** this DBA work
DBAs are often the hardest resource for IT to **justify** because they are invisible
DBAs are already being asked to do a **tremendous** amount
DBAs are often on call 24x7
DBAs are **foundational**.

I spent half my time as a DBA
Then enterprise manager
I wanted a better solution
Didn't want to automate bad process

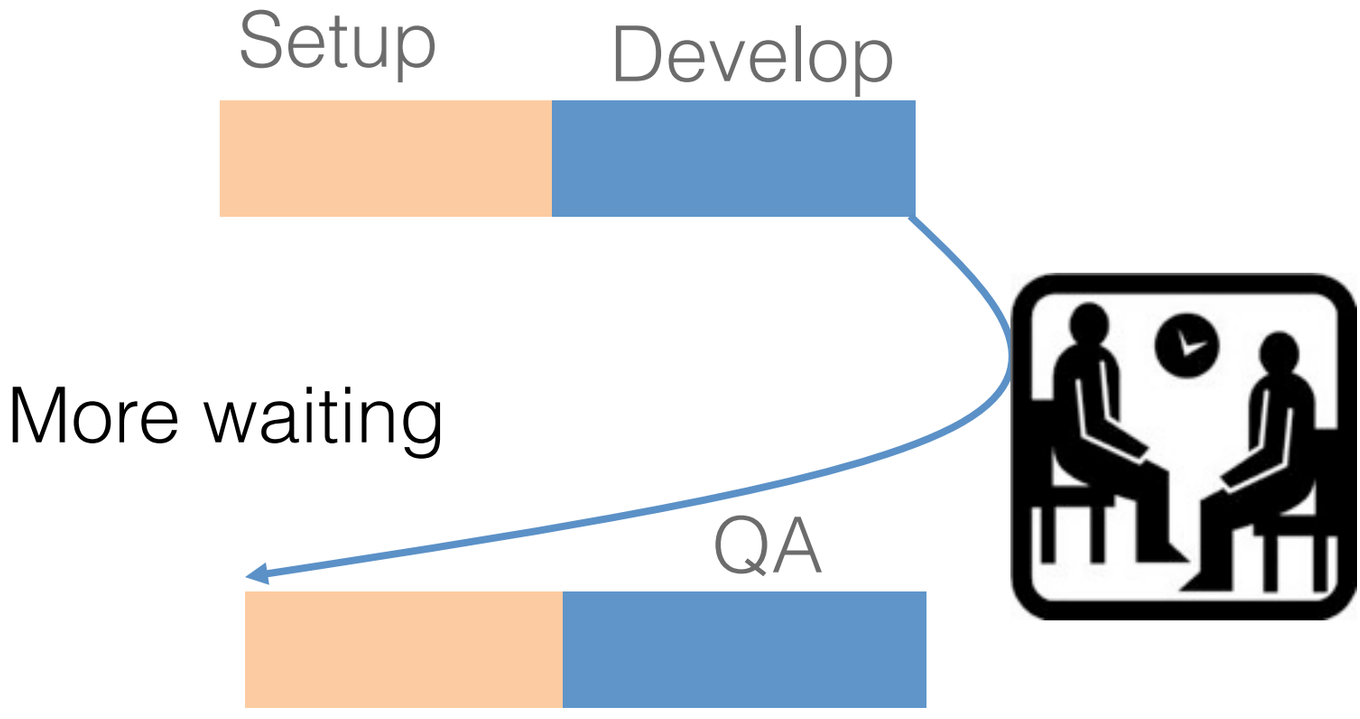
Development asks for a database it takes days or weeks.

Waiting

Setup

DELPHIX®

90% of **lost** developer days at customer was due to **waiting** for environment builds



Happens both for dev and QA

Can only support limited streams of development

Only having enough equipment to support 2 or 3 environments causes massive delays

State of Colorado has a 100 projects And they can only support 3 at a time

KLA tencor can only support 2 projects of a dozen

All this leads to bottlenecks

DELPHIX®

Slow downs mean bottlenecks

ERP Project Failures 2011

- NYC CityTime : $63 budget now $760 million.
- Montclair State University: sues Oracle PeopleSoft
- Idaho Auditors : ERP software woes cost millions

Standish Findings of IT Project Failure Rate

1994	1996	1998	2000	2002	2004	2009
31%	40%	28%	23%	15%	18%	24%

*http://www.galorath.com/wp/software-project-failure-costs-billions-better-estimation-planning-can-help.php*http://www.pcworld.com/article/246647/10_biggest_erp_software_failures_of_2011.html

ERP Failures in 2011*

NYC CityTime Project: $63 budget, total estimates reaching $760 million.
Ingram Micro: problems with an SAP project made a significant dent in profits
Montclair State University sues Oracle over a PeopleSoft project
Epicor sued by customer over ERP project that turned into a 'big mess'
State Auditors estimate ERP software woes could cost Idaho millions

*http://www.pcworld.com/article/246647/10_biggest_erp_software_failures_of_2011.html

Cost of IT Failures

Worldwide cost about USD 6.18 trillion / year

REGION	GDP (B USD)	Cost of IT Failure (B
World	69,800	6,180
USA	13,840	1,125
UK	2,260	200
Texas	1,250	110
New Zealand	44	4

2009 66% of all Federal IT dollars are projects that are at risk.

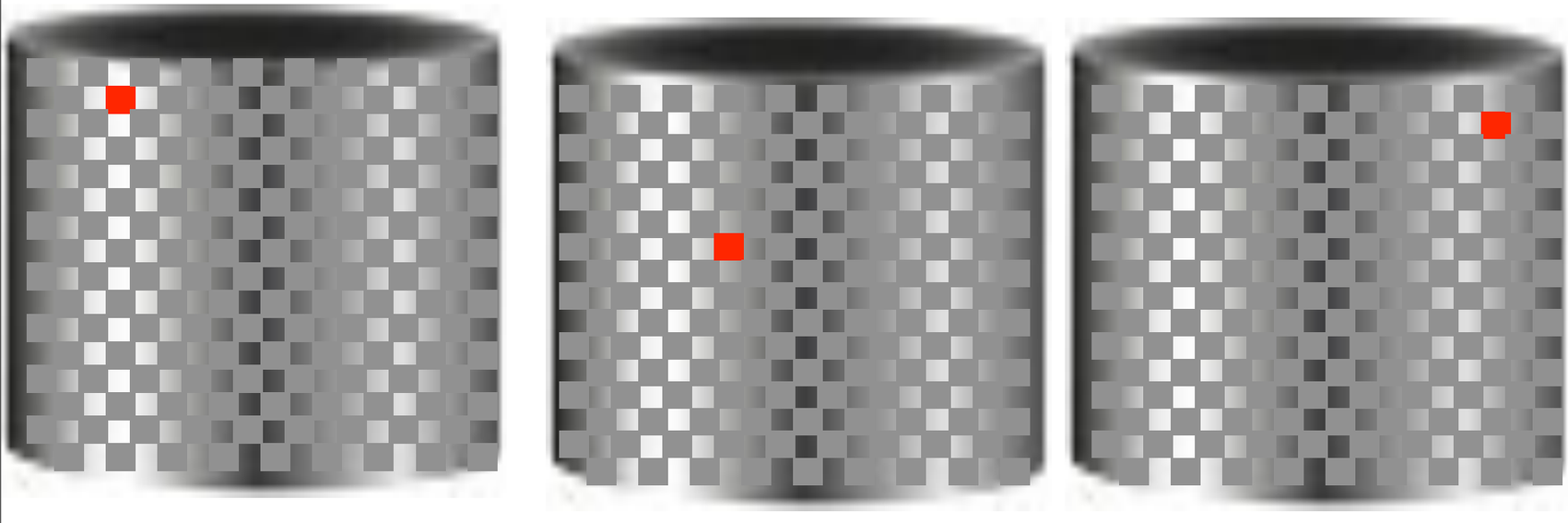
http://www.objectwatch.com/whitepapers/ITComplexityWhitePaper.pdf

99% of blocks are Identical

Clone 1

Clone 2

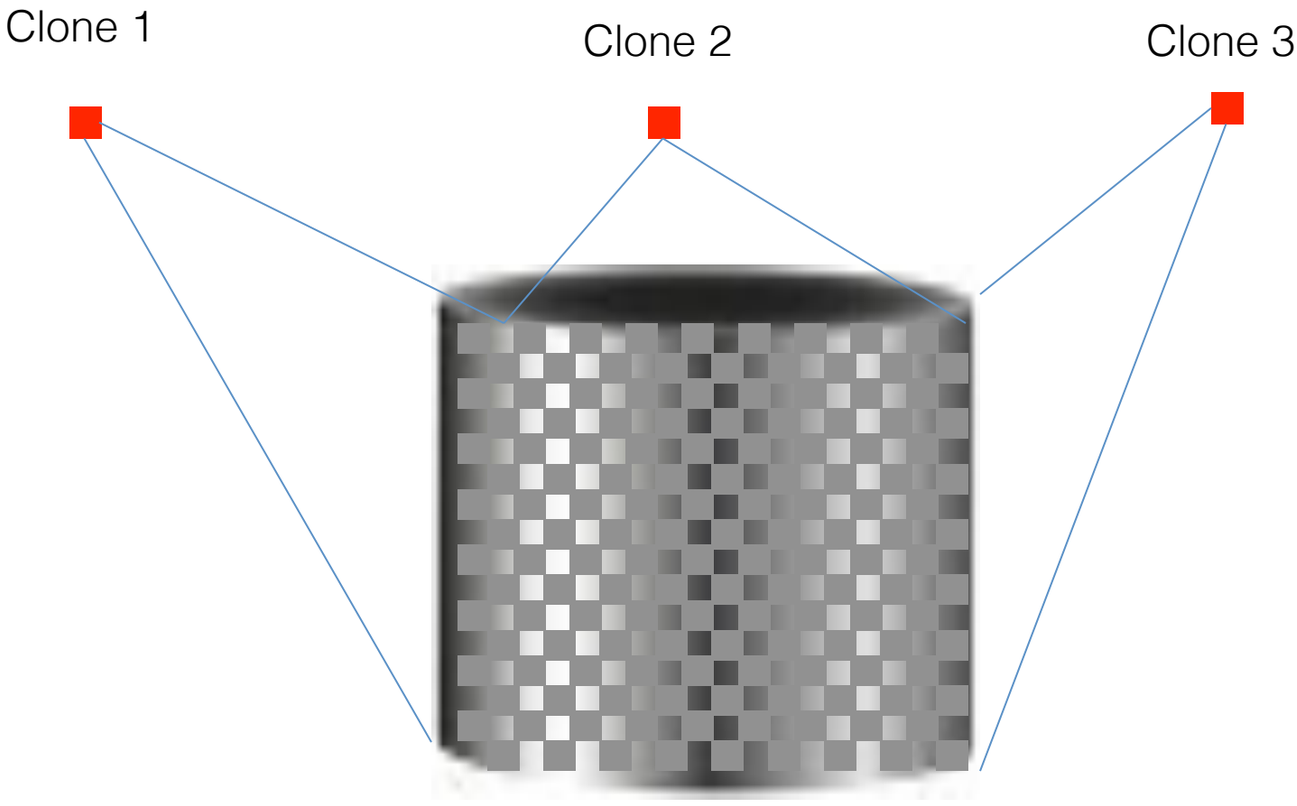
Clone 3



DELPHIX®

http://us.123rf.com/400wm/400/400/mipan/mipan1009/mipan100900010/7788876-suitcase-with-wheels.jpg

Thin Provision



DELPHIX®

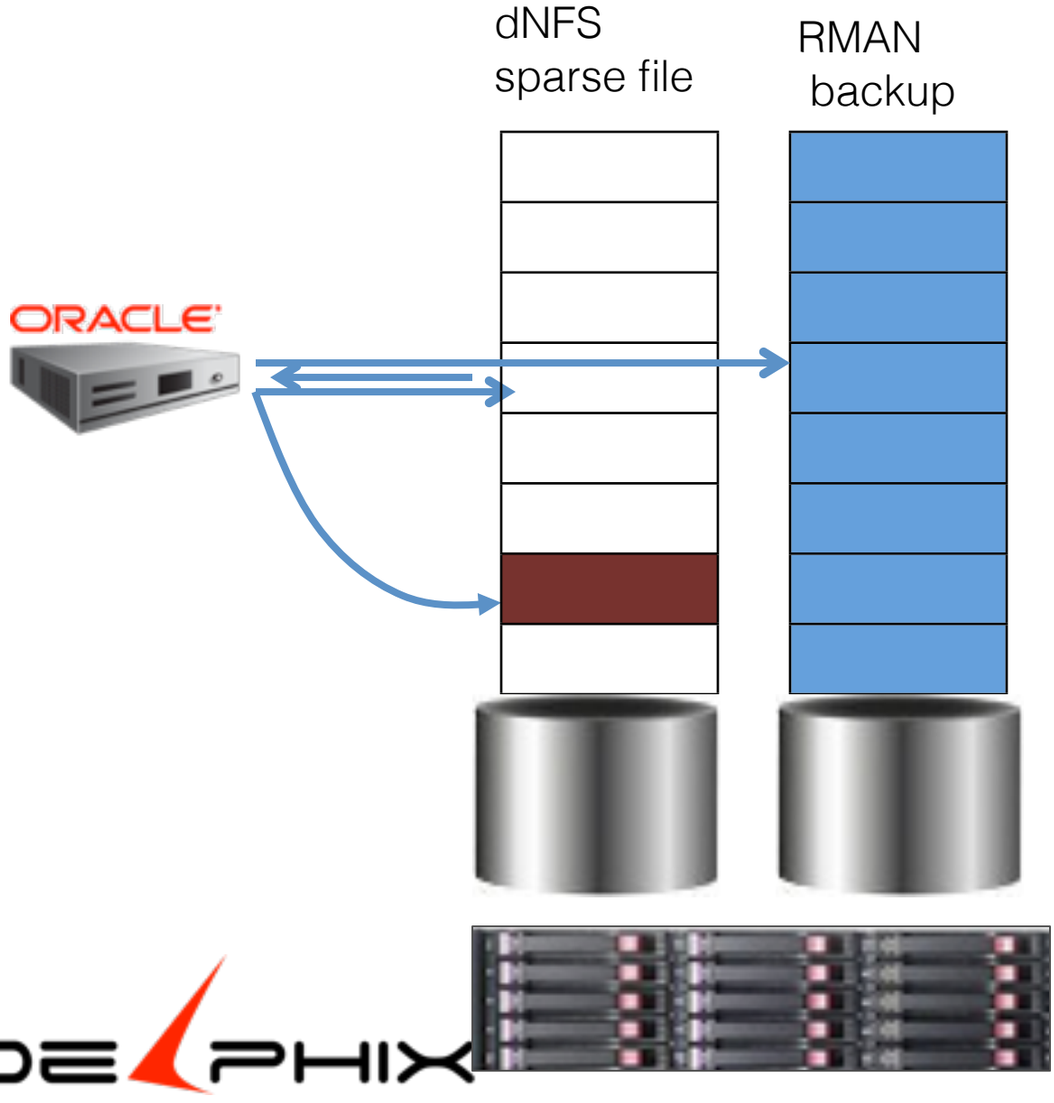
Thin Provision Cloning

Requires Cornerstone

Corner stone technology
Copy on write

1. Overlay
   - clonedb
2. Copy on Write
   - EMC (Fujitsu, IBM)
   - Vmware Data Director
3. Allocate on Write
   - Netapp
   - ZFS
   - DxFS

1. Overlay



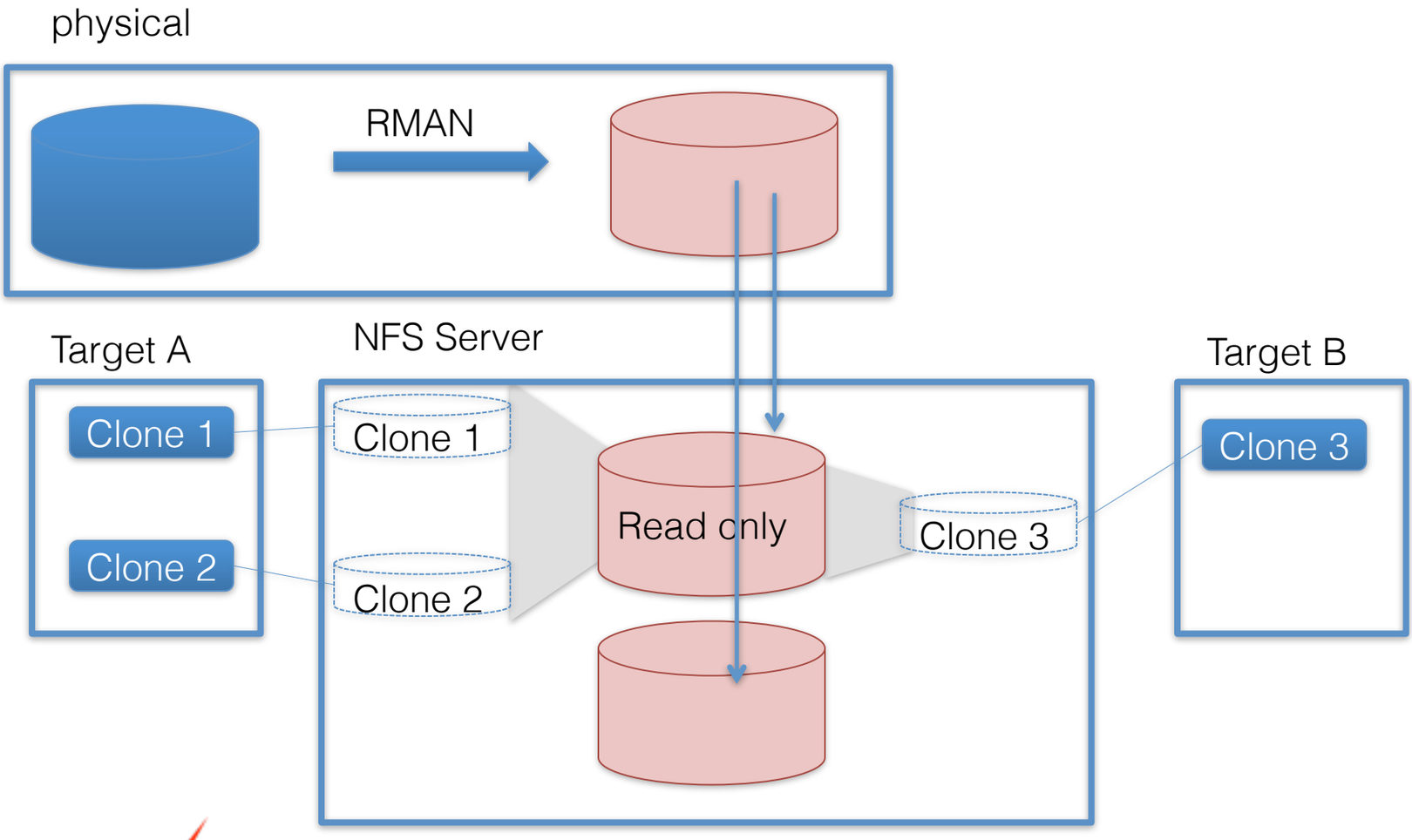
Performance issues
Single point in time

CloneDB

- **dNFS 11.2.0.2+**
  - libnfsodm11.so
  - /etc/oranfstab
- **`Clonedb.pl initSOURCE.ora output.sql`**
  - MASTER_COPY_DIR="/rman_backup”
  - CLONE_FILE_CREATE_DEST="/nfs_mount”
  - CLONEDB_NAME="clone"
- **`sqlplus / as sysdba @output.sql`**
  - startup **nomount** PFILE=initclone.ora , **Create control** file backup location
  - dbms_dnfs.**clonedb_renamefile**('/**backup**/file.dbf' , '/**clone**/file.dbf');
  - alter database **open resetlogs**;

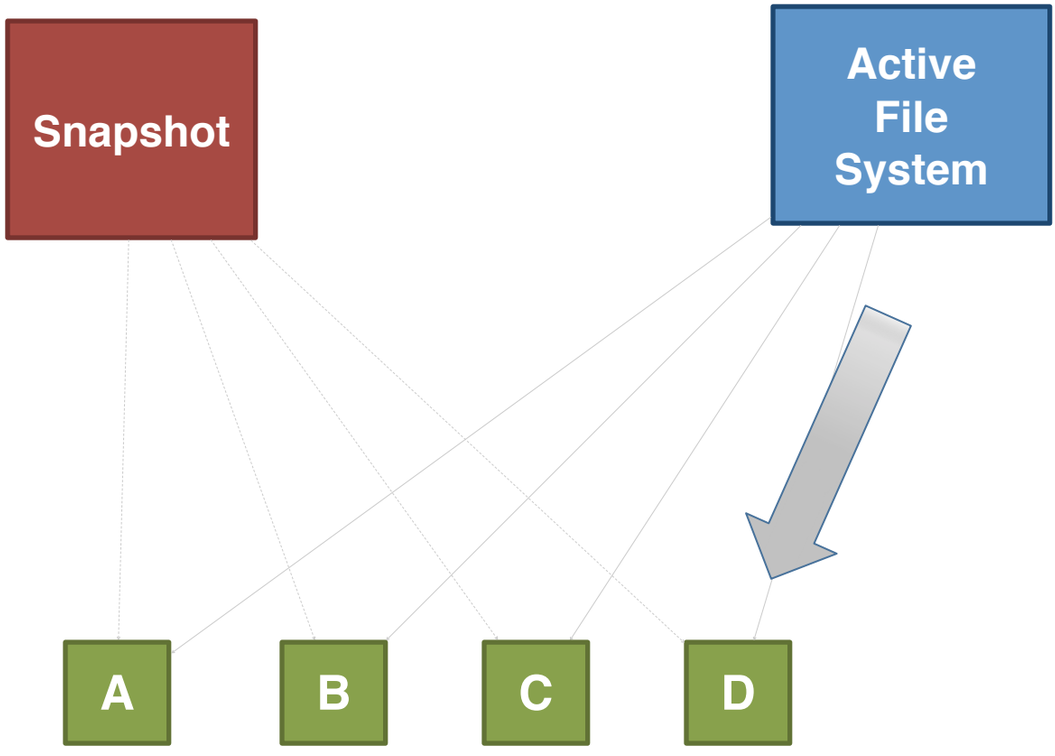
Tim Hall
www.oracle-base.com/articles/11g/

Clone DB



DELPHIX®

2. Copy on Write

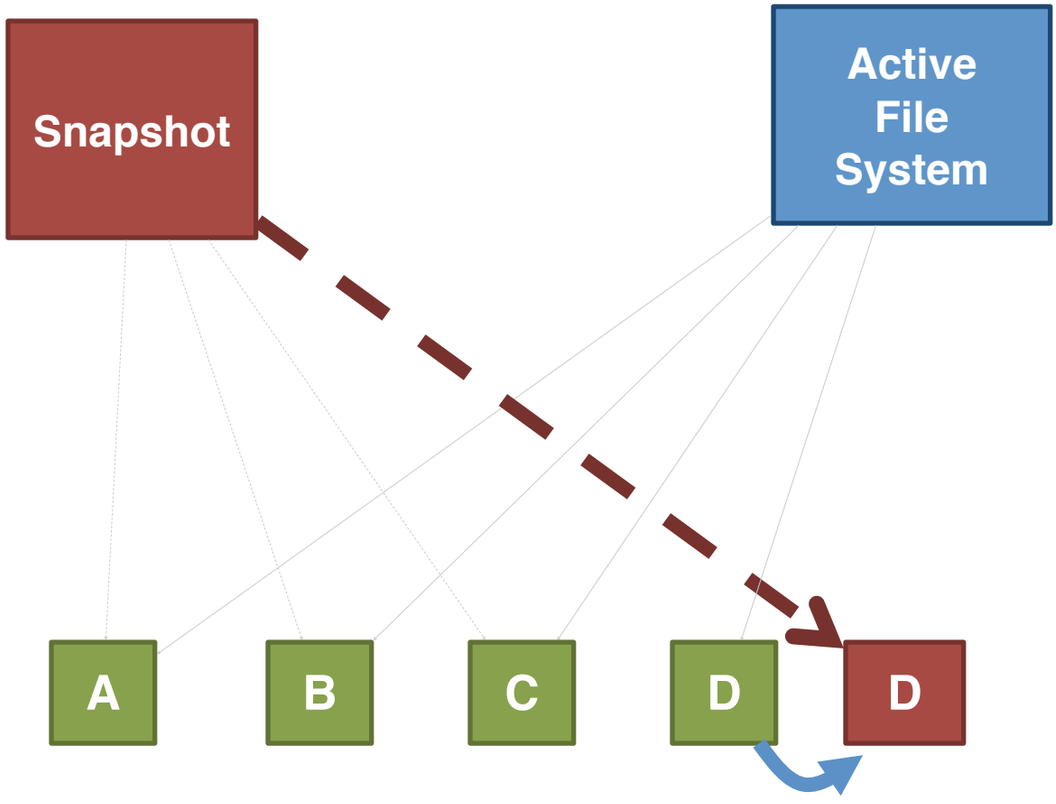


File System constant - EMC

DELPHIX

Physical File System:
Performance issues
Multiple points in time
Occasional rebuild

2. Copy on Write



Changes written in place - EMC
Changes to new area - VMware

DELPHIX®

Physical File System:
Performance issues
Multiple points in time
Occasional rebuild

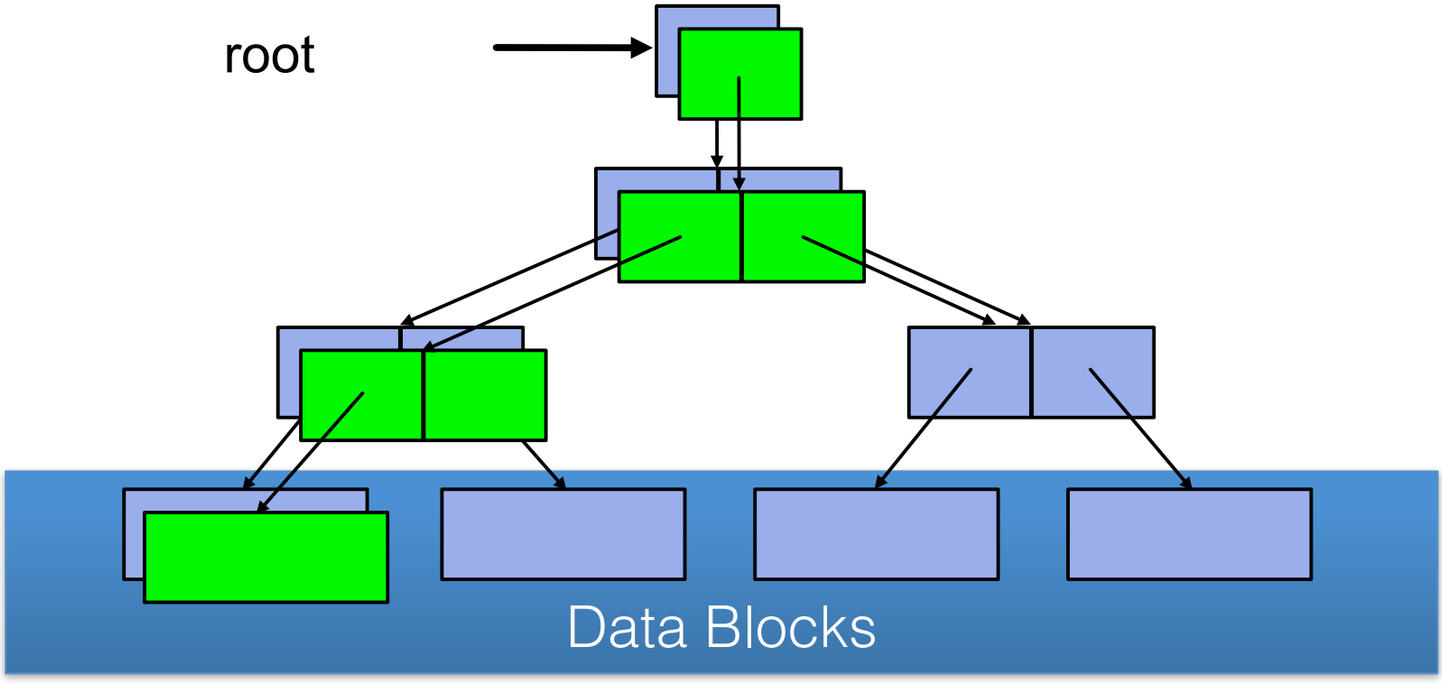
EMC

- Goal: backup
  - Create BCV , mirror , take snaps of BCV
  - Zone and mask LUN to target host
  - Full copy of disk, snapshot, recover
- “Golden Copy”
  - EMC uses a save area, area for changes to the snapshot
  - Initial snapshot has to stay
- Snapshots
  - 16 snaphshots and then have to rebuild
  - Can only take one level snap of a snap on high

Data Director : Linked Clones

- Not support for Oracle databases
- Golden Copy
- x86 host databases only
- Performance issues
  - “Having several linked clones can affect the performance of the source database and the performance of the linked clones.” http://bit.ly/QOXbyE (on http://pubs.vmware.com)
  - “If you are focused on performance, you should prefer a full clone over a linked clone.” http://www.vmware.com/support/ws5/doc/ws_clone_typeofclone.html
  - Performance worse with more snapshots
  - Performance worse with more concurrent users

3. Allocate on Write



ZFS
Netapp
Dxfs

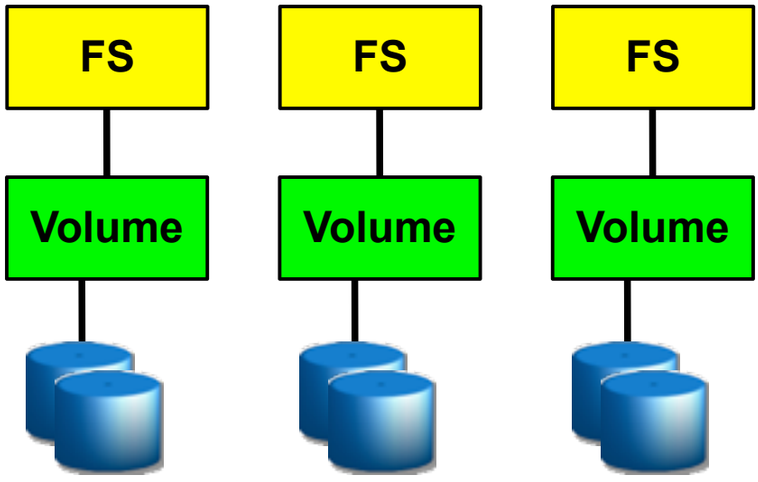
logical file system
ZFS
Netapp
Dxfs

ZFS

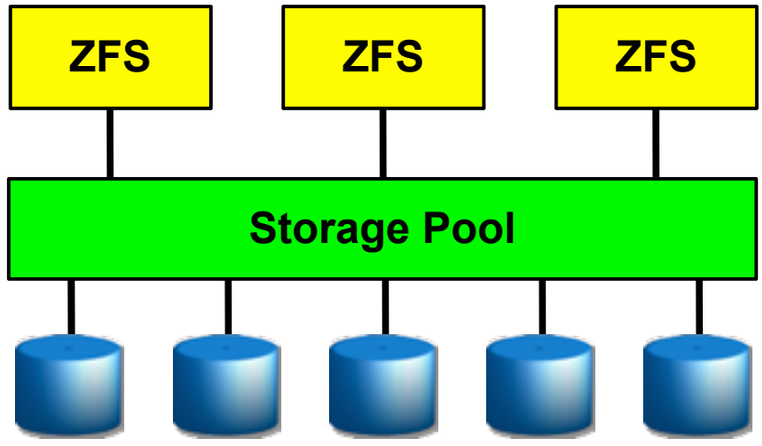
- 1 disk = 1 filesystem
- ~1990: N disks = 1 FS
- 2001: ZFS starts
- **2005:** ZFS **ships**
- 2008: ZFS storage appliance ships
- 2010: Delphix moves from ZFS to DxFS

FS vs. Pooled

- FS per Volume
- FS limited bandwidth
- Storage stranded



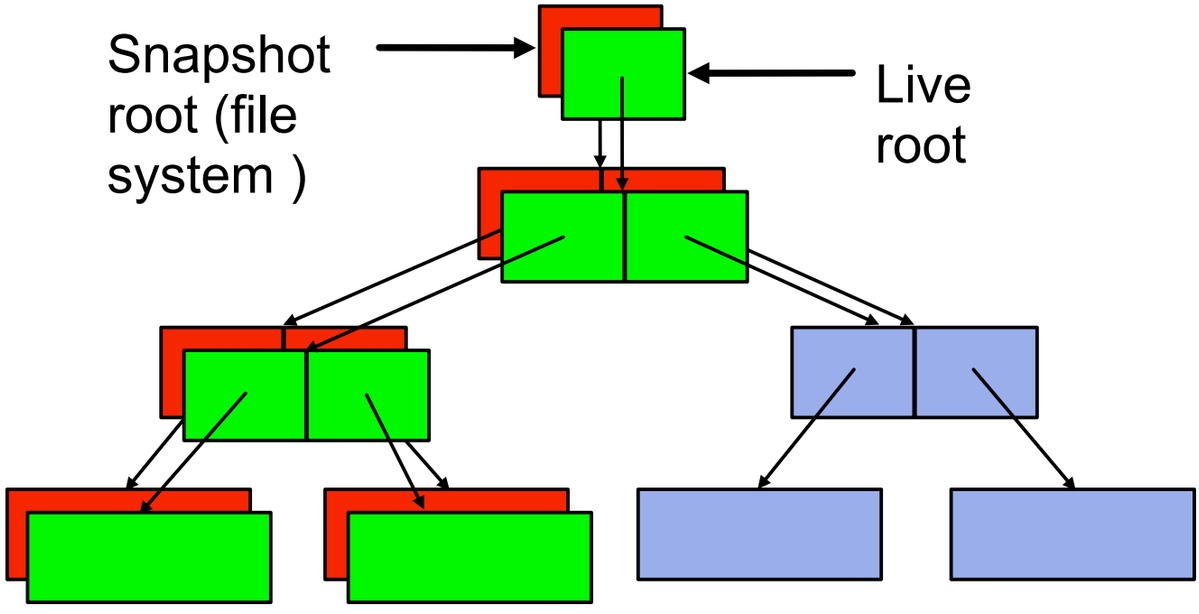
- Many FS in a pool
- Grow automatically
- All bandwidth



DELPHIX®

Bonus: Constant-Time

- No younger snapshots => free, else keep



Sync writes are written immediately out to
Intent log
Data and Metadata
Is batch written out later

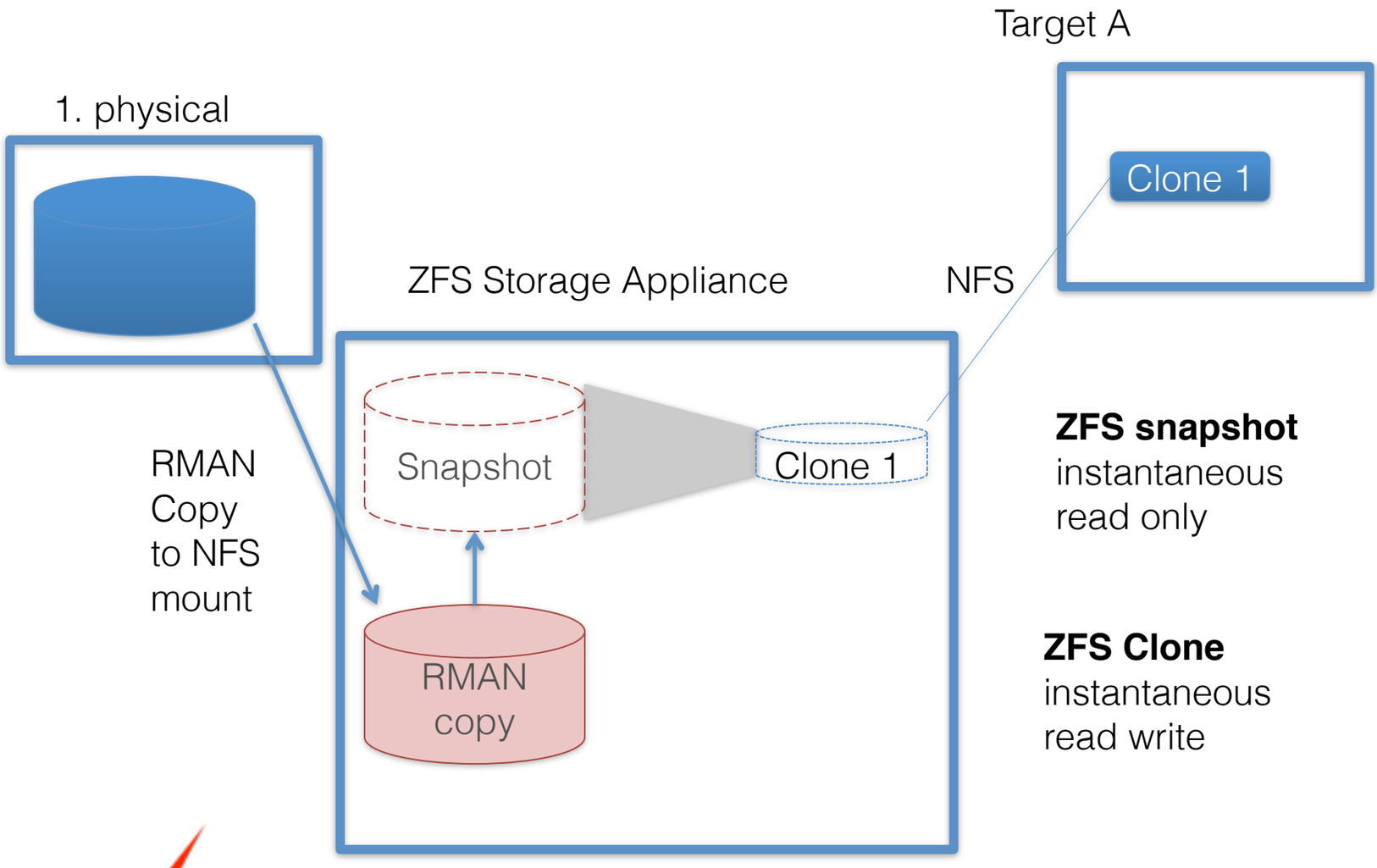
DELPHIX

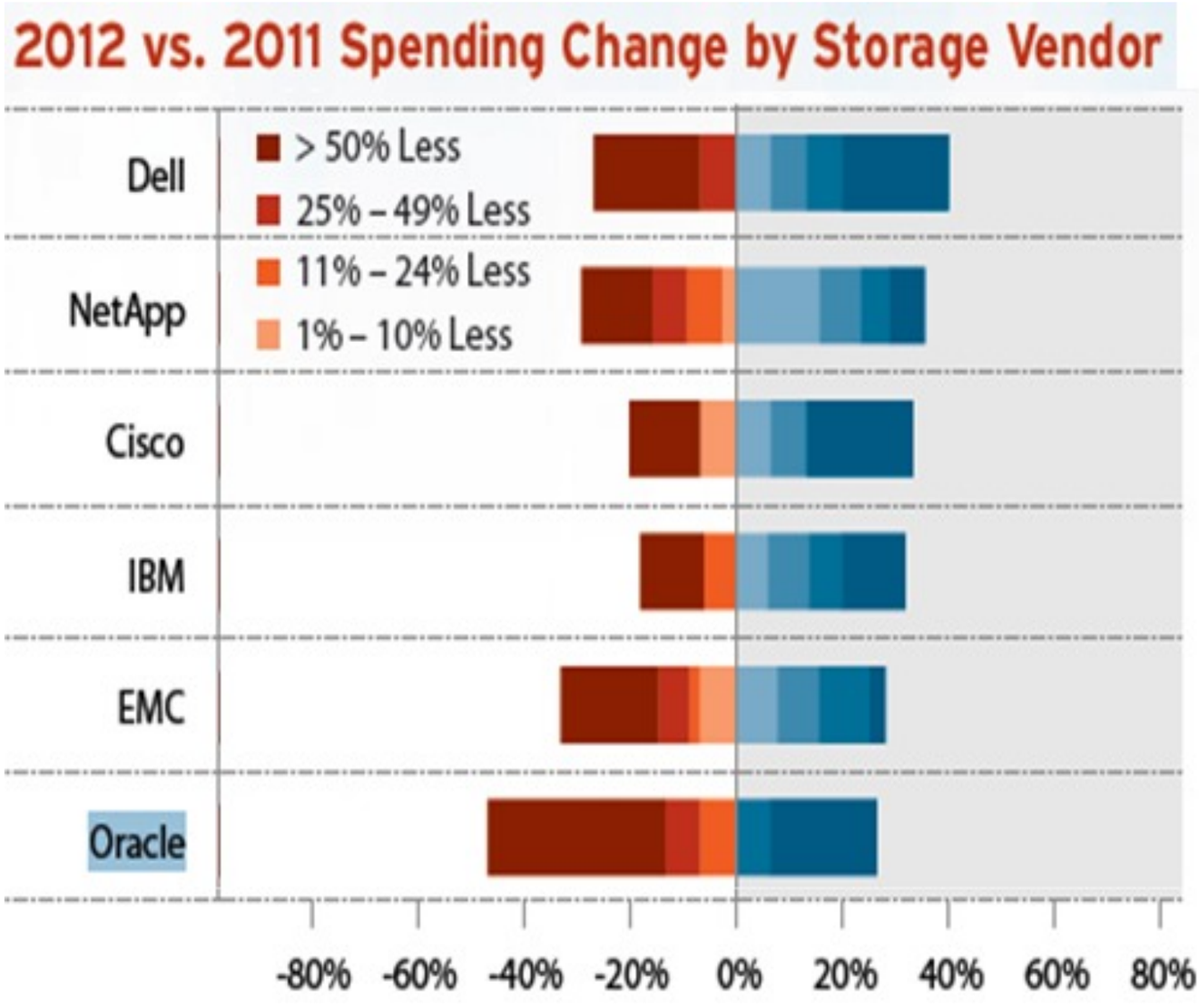
ZFS Appliance

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

cloning-solution-353626.pdf
44 pages only partial solution

Oracle ZFS Appliance





DELPHIX®

451research.com

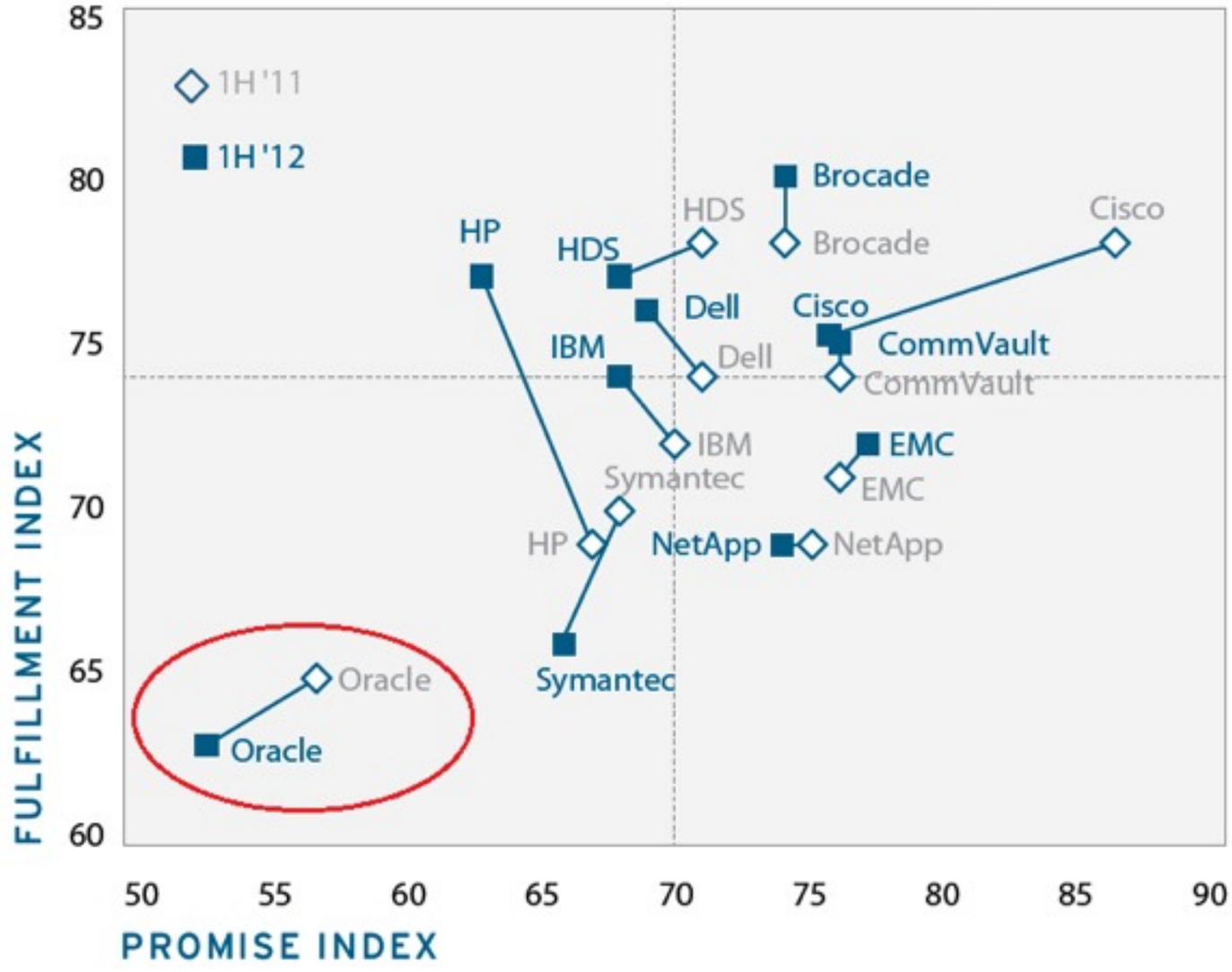
https://451research.com/report-long?icid=2556

Attached are a couple of graphics. Oracle looks pretty sad and lonely in the bottom left ratings quadrant. Oracle looks bleak as well in people pulling their money out of Oracle storage. Here are 3 quotes:

"Oracle continues to have some success with Exadata, but these reviewers are not new clients. In addition, it is still failing to shore up decreased spending as Sun and StorageTek clients leave."

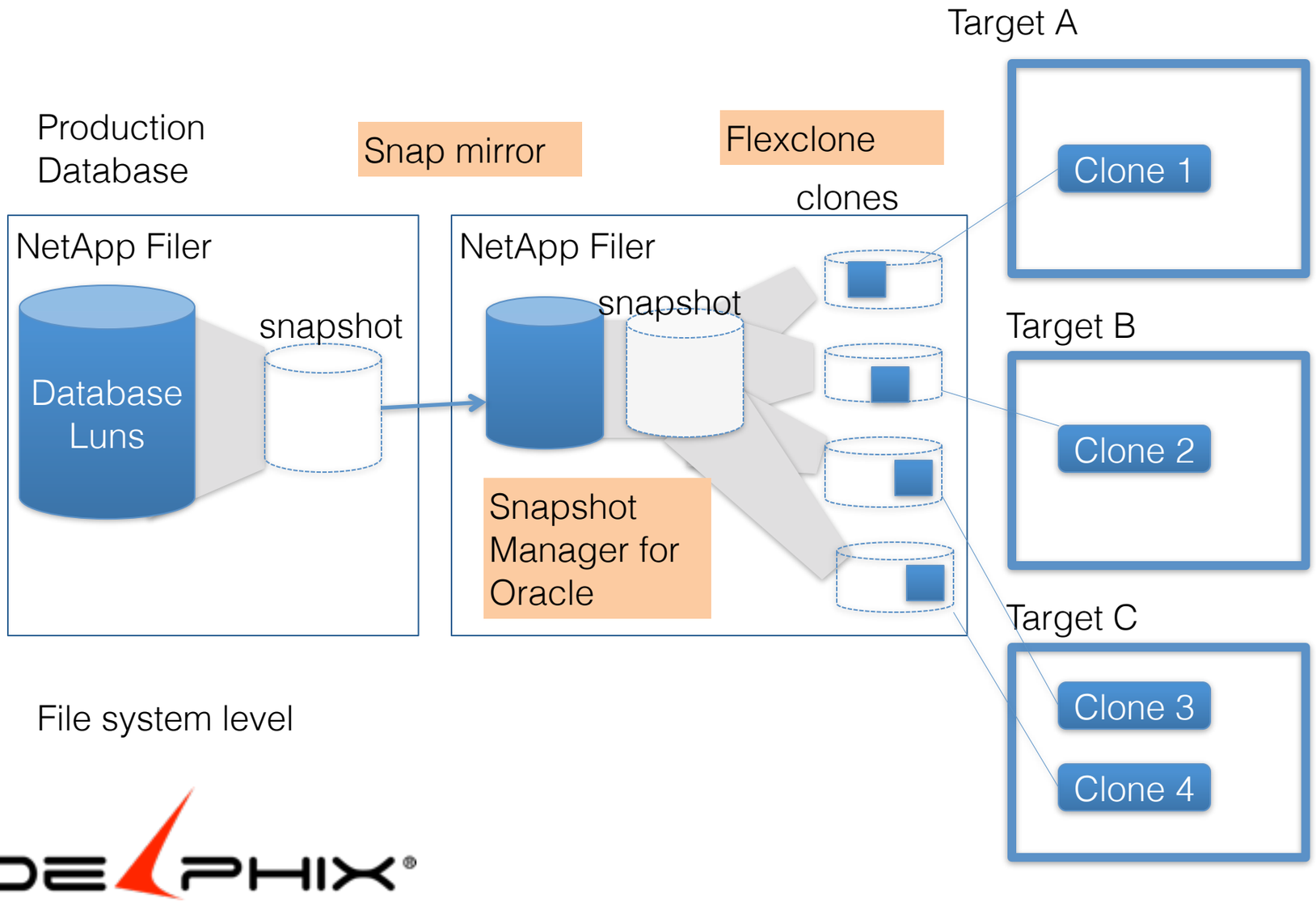
"Oracle has fallen on both the promise and fulfillment indexes and has the poorest position in the market window of any storage vendor for this study."

"Oracle ratings appear in the bottom quintile for seven of the 14 categories. Reviewers see it lacking in strategic vision, brand/reputation, ease of doing business, sales force, technical support, product quality and reliability."


1H'11
1H'12
FULFILLMENT INDEX
PROMISE INDEX
85
80
75
70
65
60
50
55
60
65
70
75
80
85
90
Brocade
Brocade
HDS
HDS
Cisco
Cisco
HP
Dell
Dell
CommVault
CommVault
IBM
IBM
EMC
EMC
Symantec
Symantec
HP
NetApp
NetApp
Oracle
Oracle

DELPHIX®

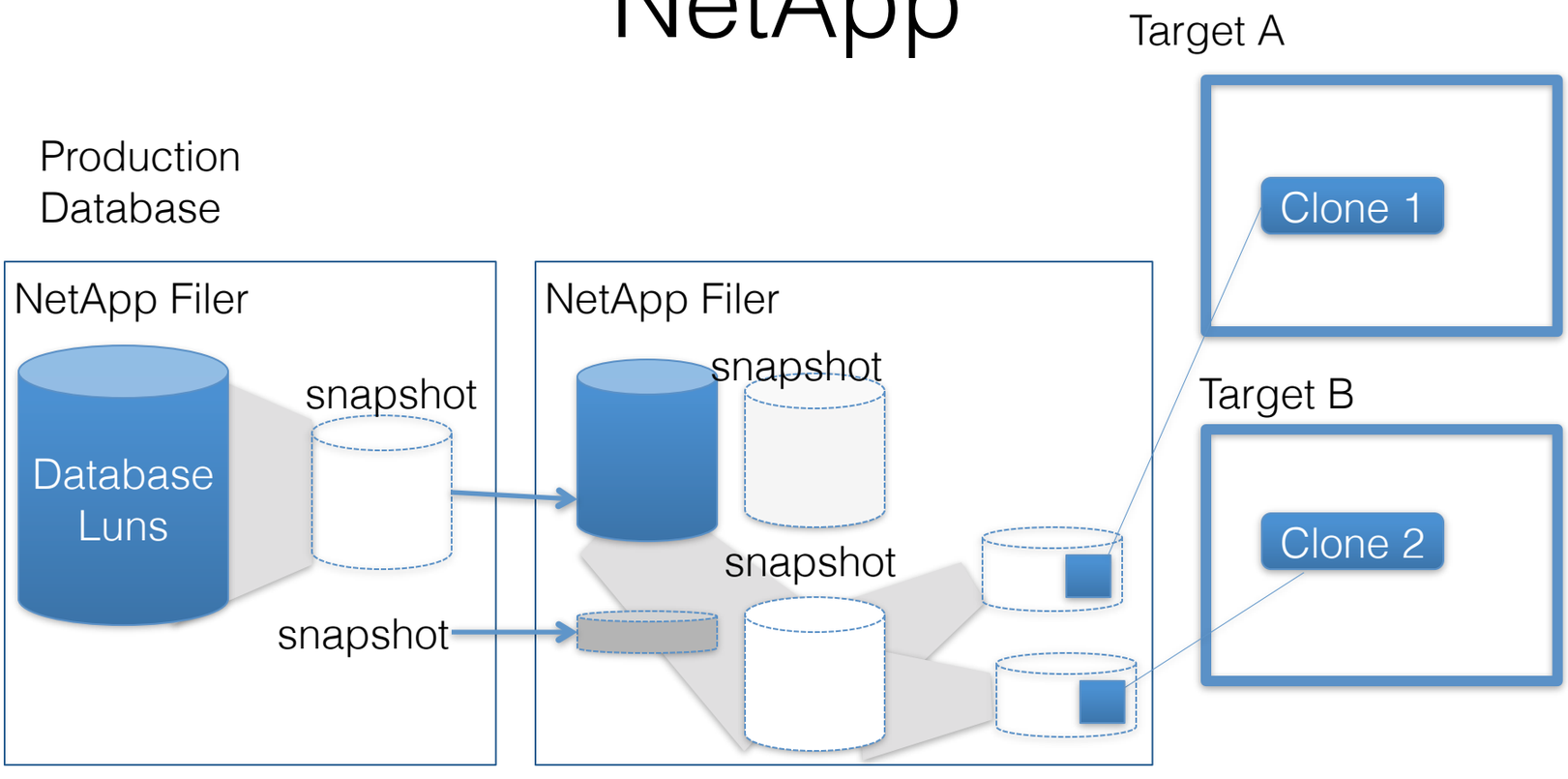
NetApp



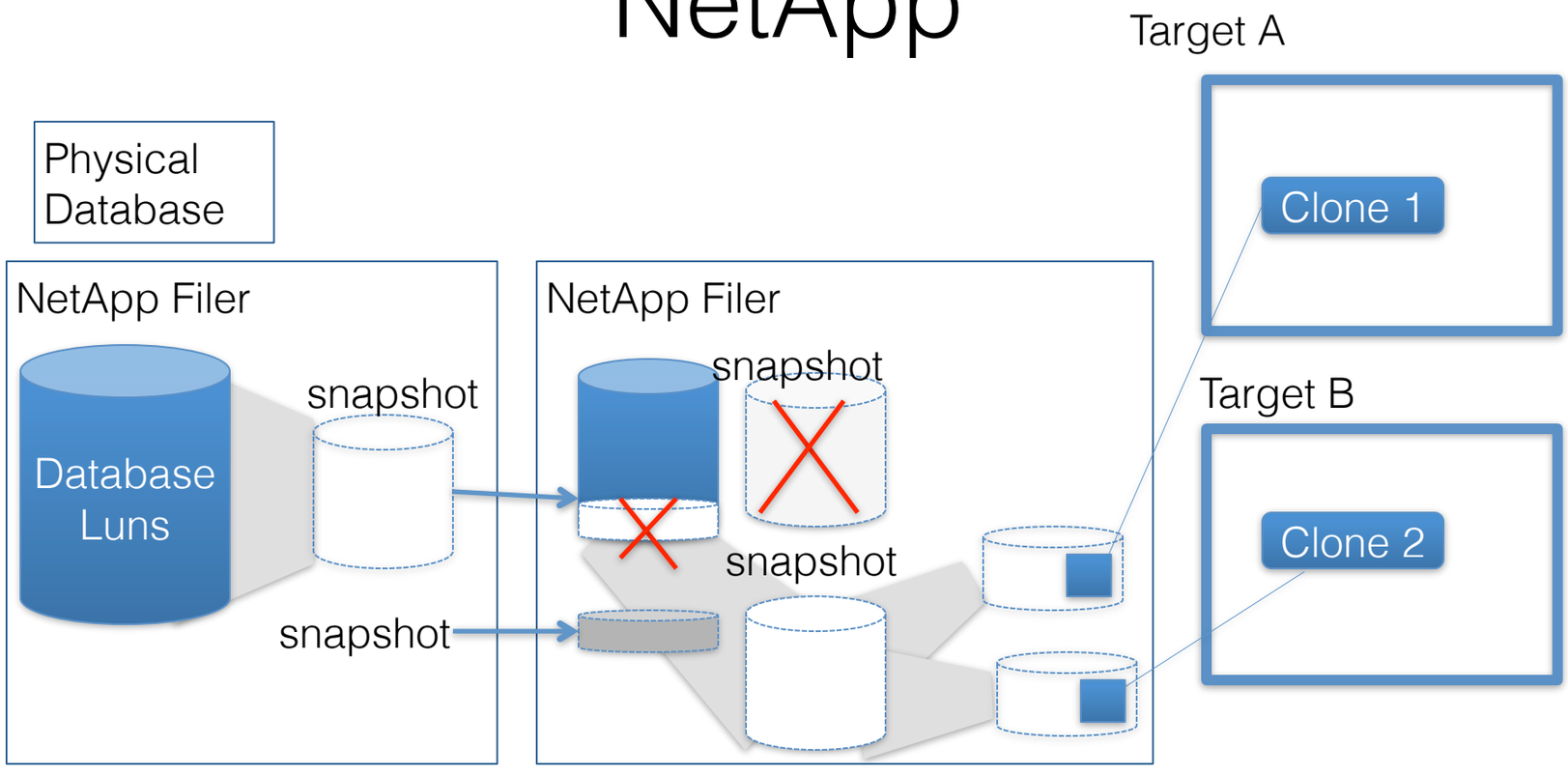
File system level

DELPHIX

NetApp


Target A
Production
Database
Clone 1
NetApp Filer
NetApp Filer
snapshot
Target B
Database
Luns
snapshot
snapshot
Clone 2
snapshot

NetApp


Target A
Physical Database
Clone 1
NetApp Filer
NetApp Filer
snapshot
Target B
Database Luns
snapshot
Clone 2
snapshot
snapshot

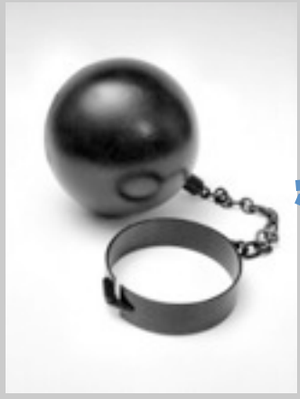
NetApp Limits

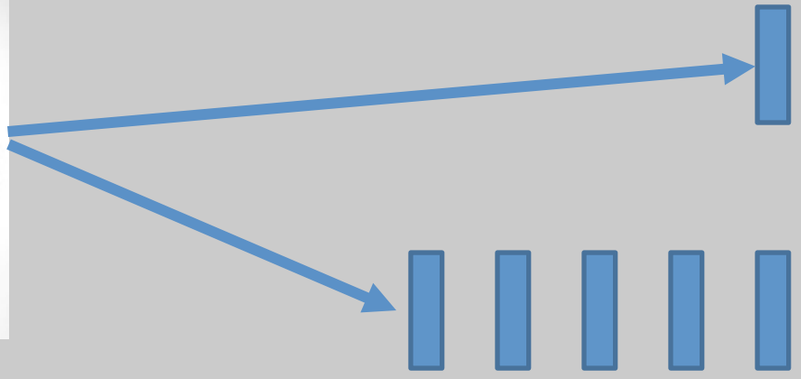
Limit of 255 snapshots
snaps are limited to the same aggregate (storage pool)
Aggregates have size limits depending on controller

Controller	Size Limit
32 bit controllers	16TB
FAS3140/FAS3040/FAS3050	40TB
FAS3160/FAS3070	50TB
FAS6040/FAS3170	70TB
FAS6080	100TB

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

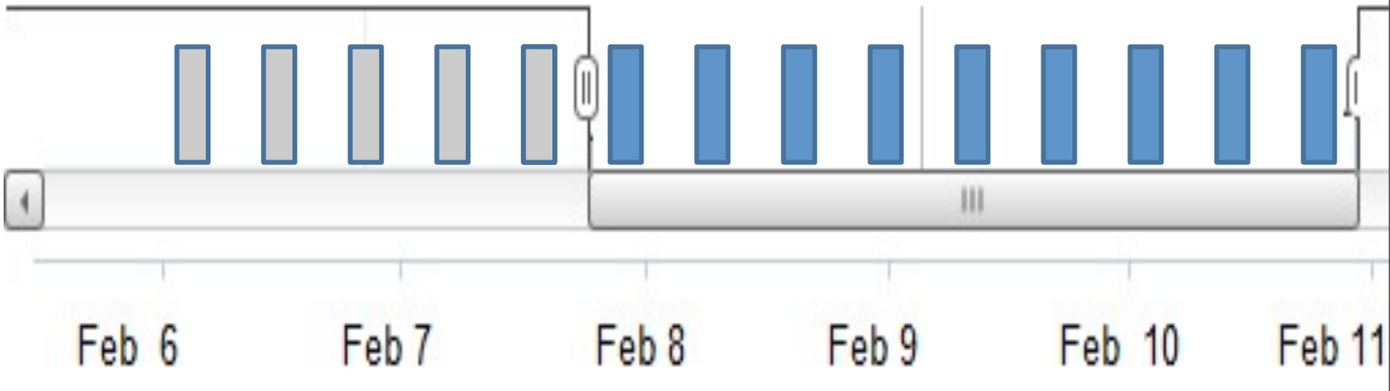
Clonedb





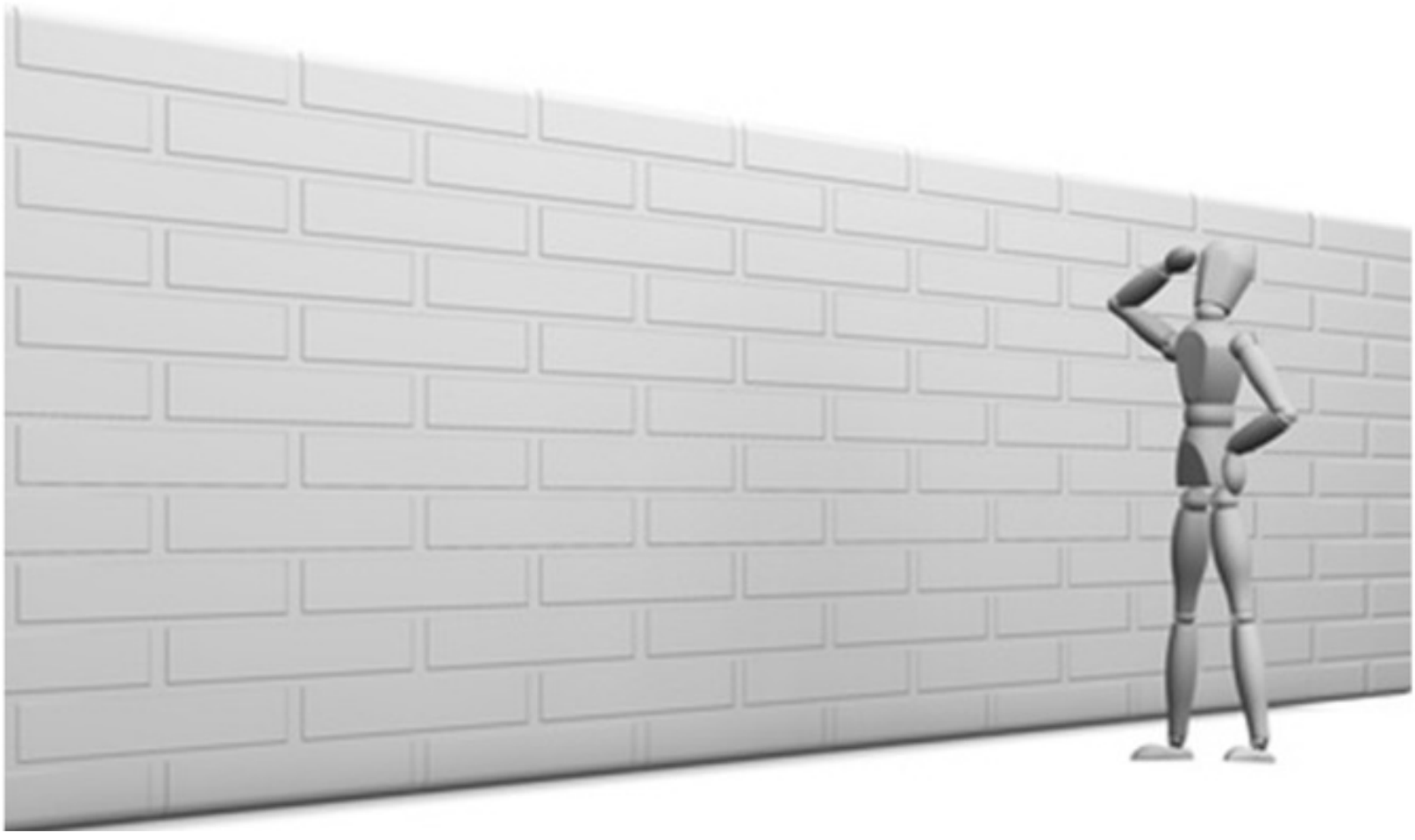
EMC

Netapp



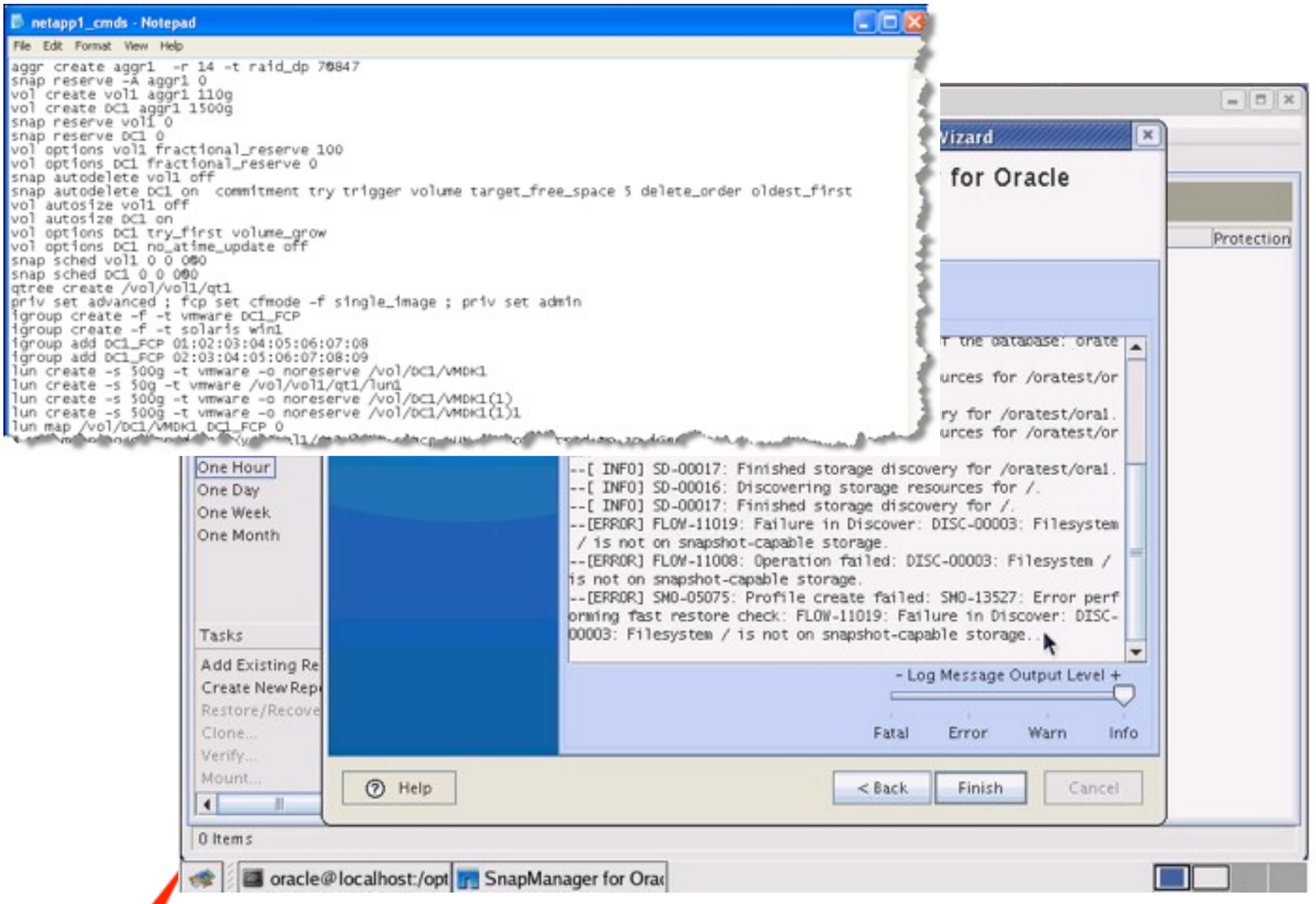
DELPHIX®

1. Overlay
   - Clonedb - 1 copy, performance issues
2. Copy on write
   - EMC - 16 points in time performance issues
   - VMware – not support with Oracle
3. Allocate on Write
   - ZFS – manually config, performance issues
   - NetApp – 255 points in time , rolling window

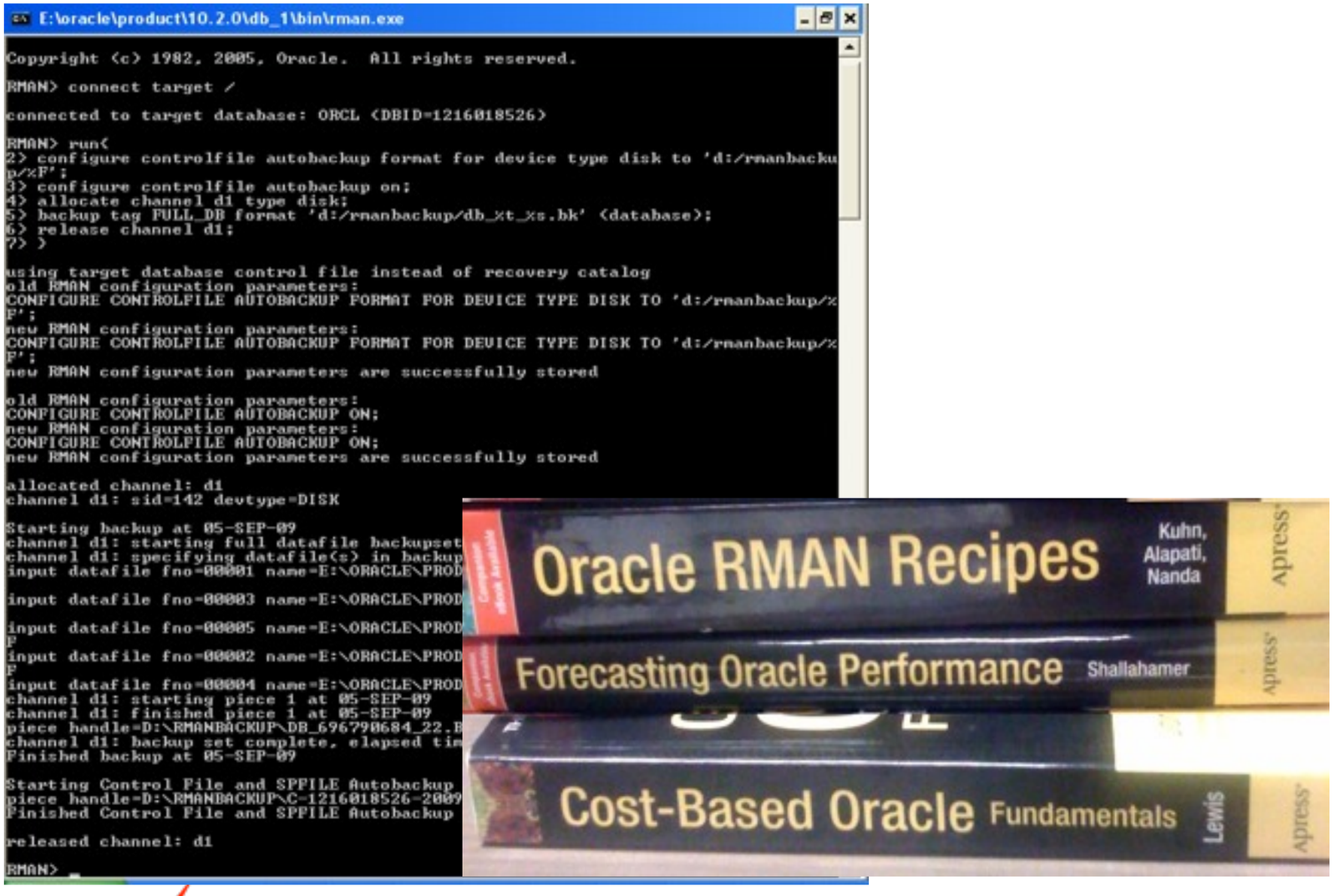


DELPHIX

Technology has existed 15+ years
Why hasn't there been more adoption ??



http://partners.netapp.com/go/techontap/empower-dba.html?fmt=print
Create Luns, aggr, snapshots, clones
Mirroring filesystems
Exporting file systems
Mounting file systems


E:\oracle\product\10.2.0\db_1\bin\rman.exe
Copyright (c) 1982, 2005, Oracle. All rights reserved.
RMAN> connect target /
connected to target database: ORCL (DBID=1216018526)
RMAN> run{
2> configure controlfile autobackup format for device type disk to 'd:/rmanbacku
p/%F';
3> configure controlfile autobackup on;
4> allocate channel d1 type disk;
5> backup tag FULL_DB format 'd:/rmanbackup/db_%t_%s.bk' (database);
6> release channel d1;
7> }
using target database control file instead of recovery catalog
old RMAN configuration parameters:
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK TO 'd:/rmanbackup/%
F';
new RMAN configuration parameters:
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK TO 'd:/rmanbackup/%
F';
new RMAN configuration parameters are successfully stored
old RMAN configuration parameters:
CONFIGURE CONTROLFILE AUTOBACKUP ON;
new RMAN configuration parameters:
CONFIGURE CONTROLFILE AUTOBACKUP ON;
new RMAN configuration parameters are successfully stored
allocated channel: d1
channel d1: sid=142 devtype=DISK
Starting backup at 05-SEP-09
channel d1: starting full datafile backupset
channel d1: specifying datafile(s) in backup
input datafile fno=00001 name=E:\ORACLE\PROD
input datafile fno=00003 name=E:\ORACLE\PROD
input datafile fno=00005 name=E:\ORACLE\PROD
F
input datafile fno=00002 name=E:\ORACLE\PROD
F
input datafile fno=00004 name=E:\ORACLE\PROD
channel d1: starting piece 1 at 05-SEP-09
channel d1: finished piece 1 at 05-SEP-09
piece handle=D:\RMANBACKUP\DB_696790684_22.B
channel d1: backup set complete, elapsed tim
Finished backup at 05-SEP-09
Starting Control File and SPFILE Autobackup
piece handle=D:\RMANBACKUP\C-1216018526-2009
Finished Control File and SPFILE Autobackup
released channel: d1
RMAN>
Oracle RMAN Recipes
Kuhn, Alapati, Nanda
Apress
Forecasting Oracle Performance
Shallahamer
Apress
Cost-Based Oracle Fundamentals
Lewis
Apress

DELPHIX®

Requires

expert storage admins

specialized equipment

scripting

Requires
expert storage admins
specialized equipment
scripting

CERN recently gave a presentation where they wrote almost 30,000 lines of code
13k lines & 15k lines of PHP

If it's only saving storage, storage prices are falling so of limited payoff

Thin provisioning technology ultimately **led** to database virtualization,

which goes **beyond** thin clone provisioning to

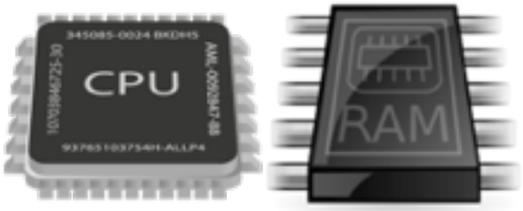
eliminate the **overhead of managing** the cloning process

providing significant **agility** gains

Thin provisioning technology ultimately **led** to database virtualization,
which goes **beyond** thin clone provisioning to
eliminate the **overhead of managing** the cloning process
providing significant **agility** gains

Database Virtualization

Virtualization Layer



vmware

DELPHIX

ORACLE

1. Oracle Snapshot Management Utility (SMU)
2. Delphix Virtual Databases

DELPHIX

Database virtualization is to the **data tier** what
VMware is to the **compute tier**.
On the compute tier
VMware allows the same **hardware** to be **shared** by multiple machines.
On the data tier virtualization allows the
same **datafiles** to be **shared** by multiple clones

allowing almost **instantaneous** creation of new copies of databases
with almost **no disk** footprint.

Oracle 12c SMU

Oracle Snap Management Utility for ZFS Appliance

- Requires ZFS Appliance
- Require Source on Appliance
- GUI
  - Takes to snapshot source databases
  - provision virtual databases

DELPHIX®

Software
installs an any x86 hardware
uses any storage
supports any Oracle OS

Automated Managed Orchestrate d

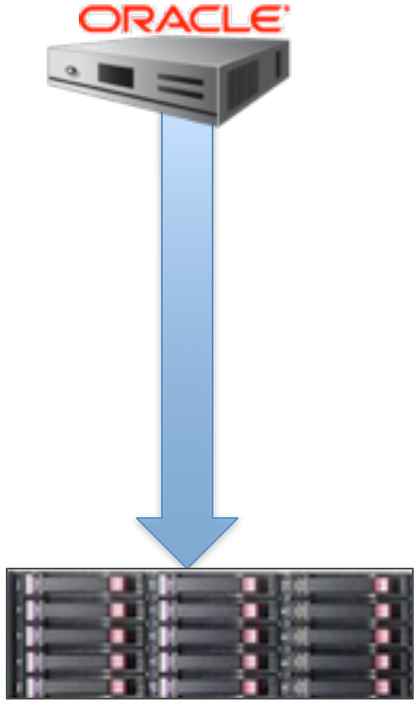
Automated
Managed
Orchestrated

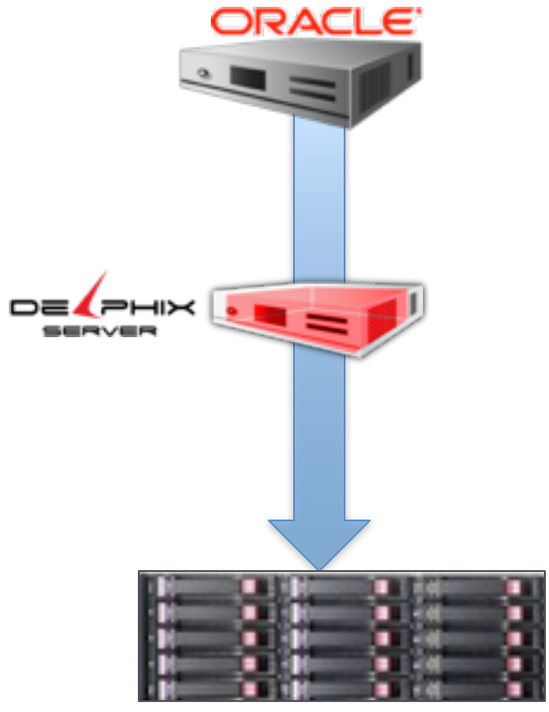
incremental forever collection

DELPHIX®

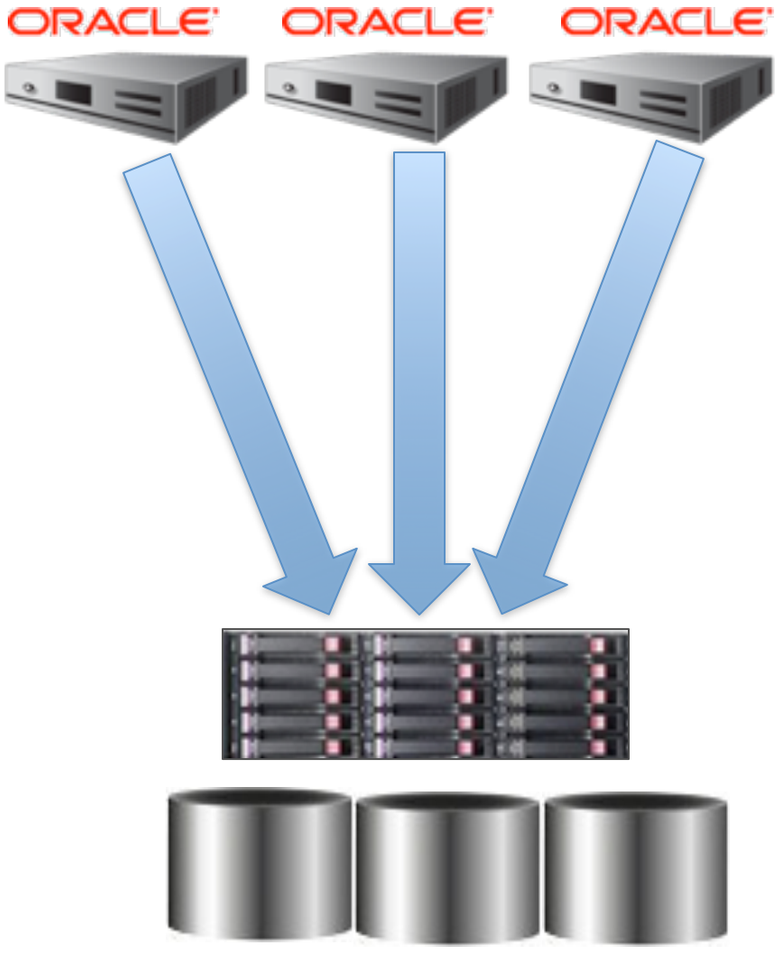
Delphix GUI is what Oracle Enterprise Manager would look like if Apple had designed it

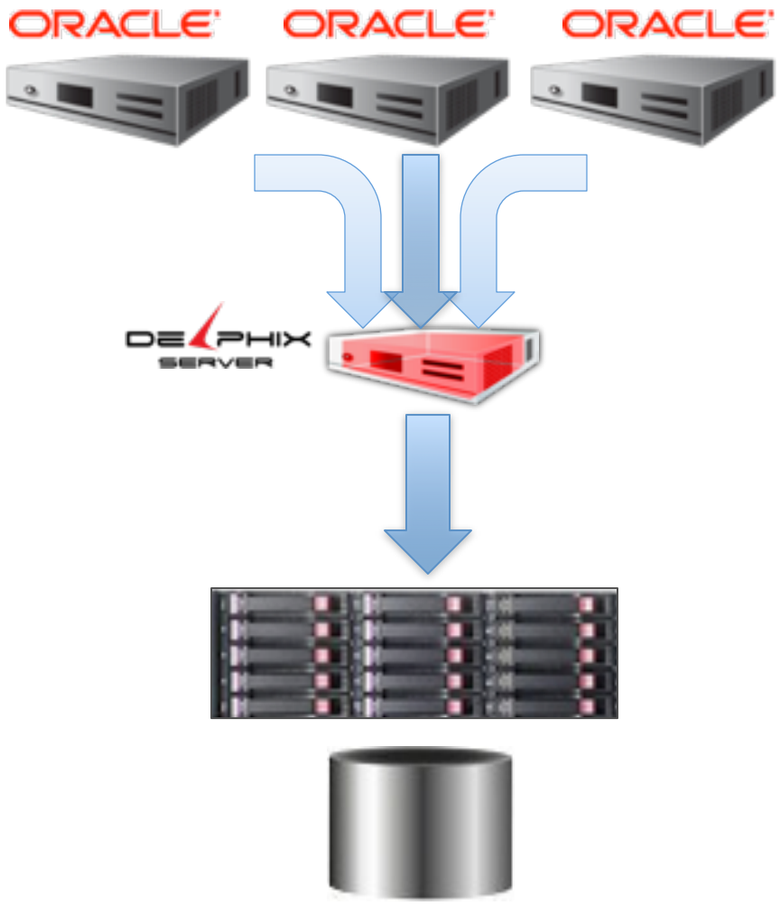
Always wanted
Steve jobs and Ellison ui combined forces
now I have it





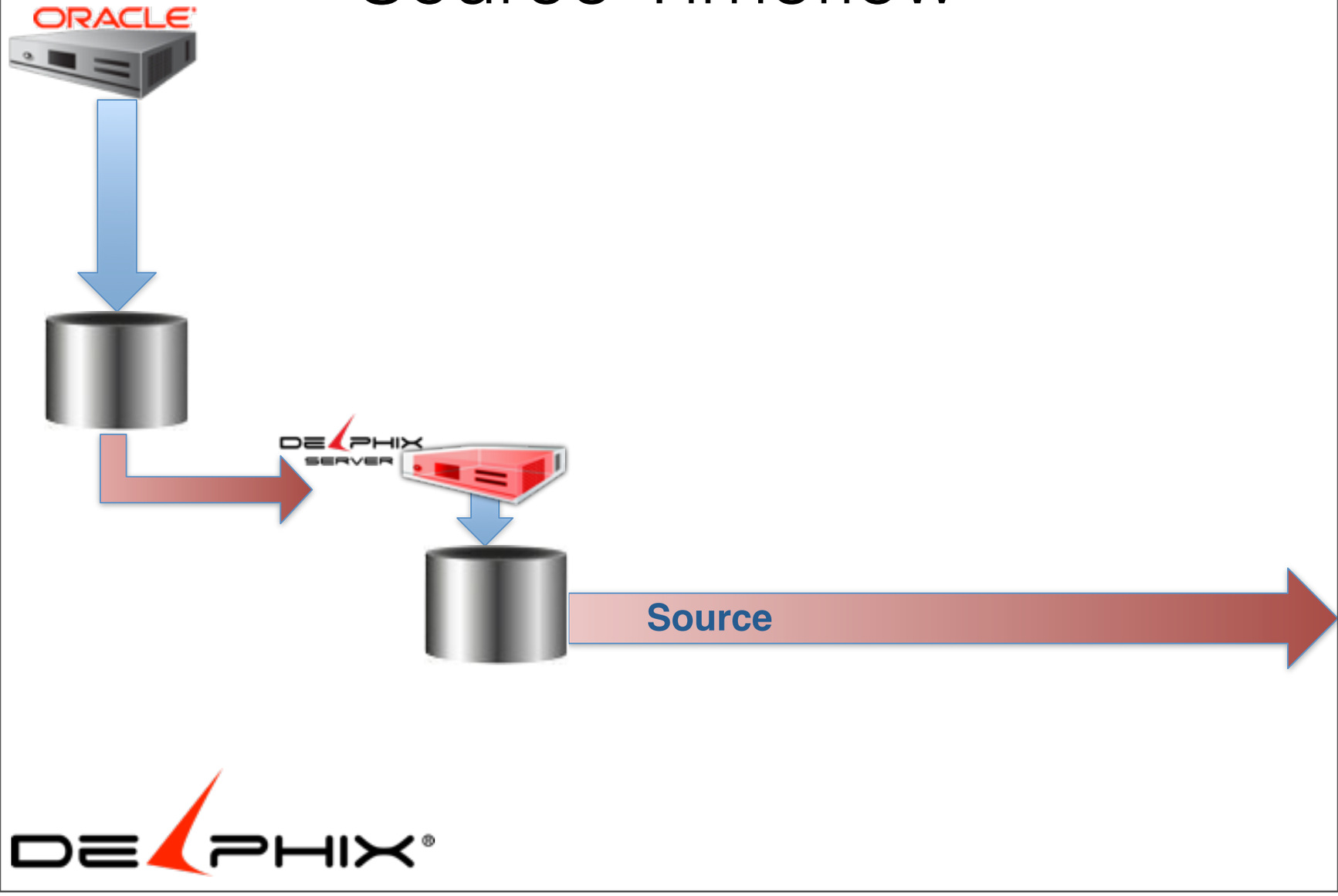
DELPHIX®





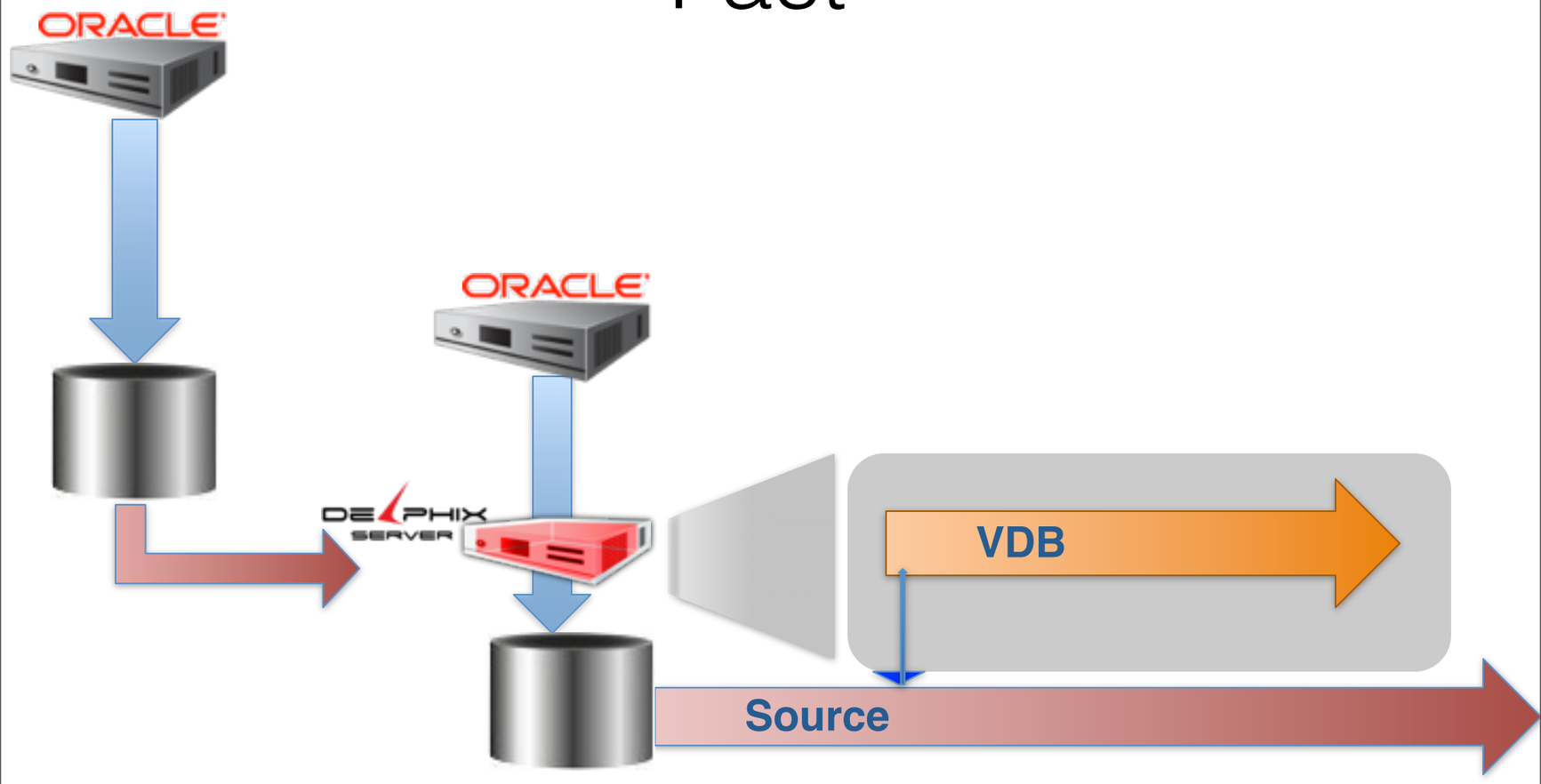
DELPHIX®

Source Timeflow



DELPHIX®

Fast



DELPHIX®

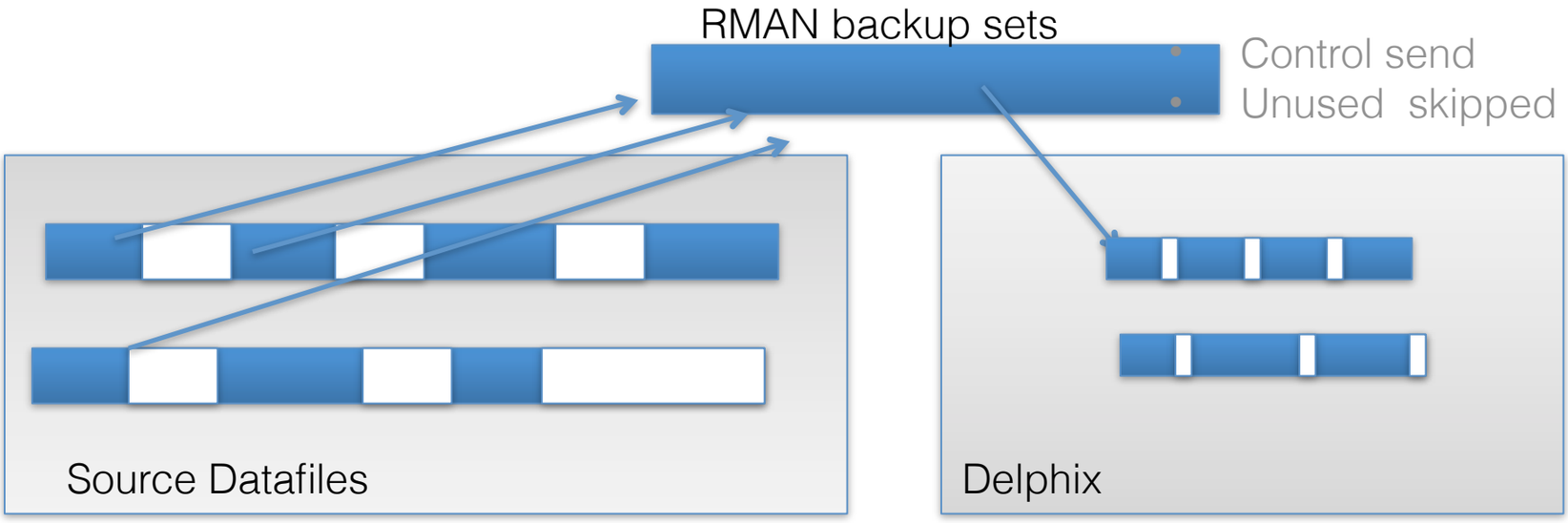
2:11 finished link
2:41 snapshot
3:30 snapshot finished
4:52 finished

DxFS at Delphix

- DxFS optimized for databases
  - Write optimizations
  - Space allocation
  - Space de-allocation and destroy
- Compression
  - typically ~2-4x
- Block sharing
  - Disk
  - Memory

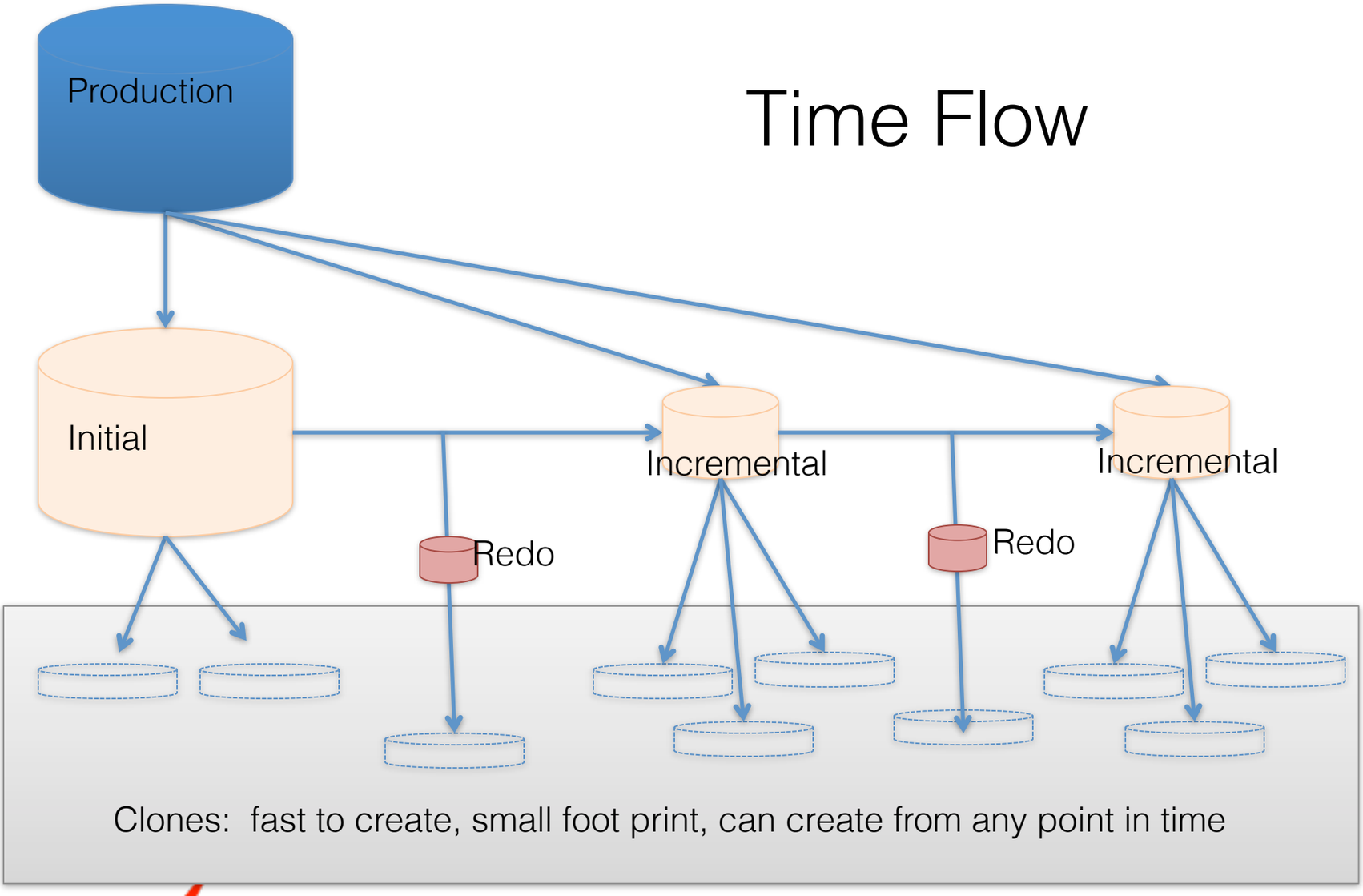
ZFS written as a general file system
DxFS optimized for databases

Delphix



- Rebuilds
- Create unused
- Compresses

Time Flow



DELPHIX®

Part II: Development Acceleration

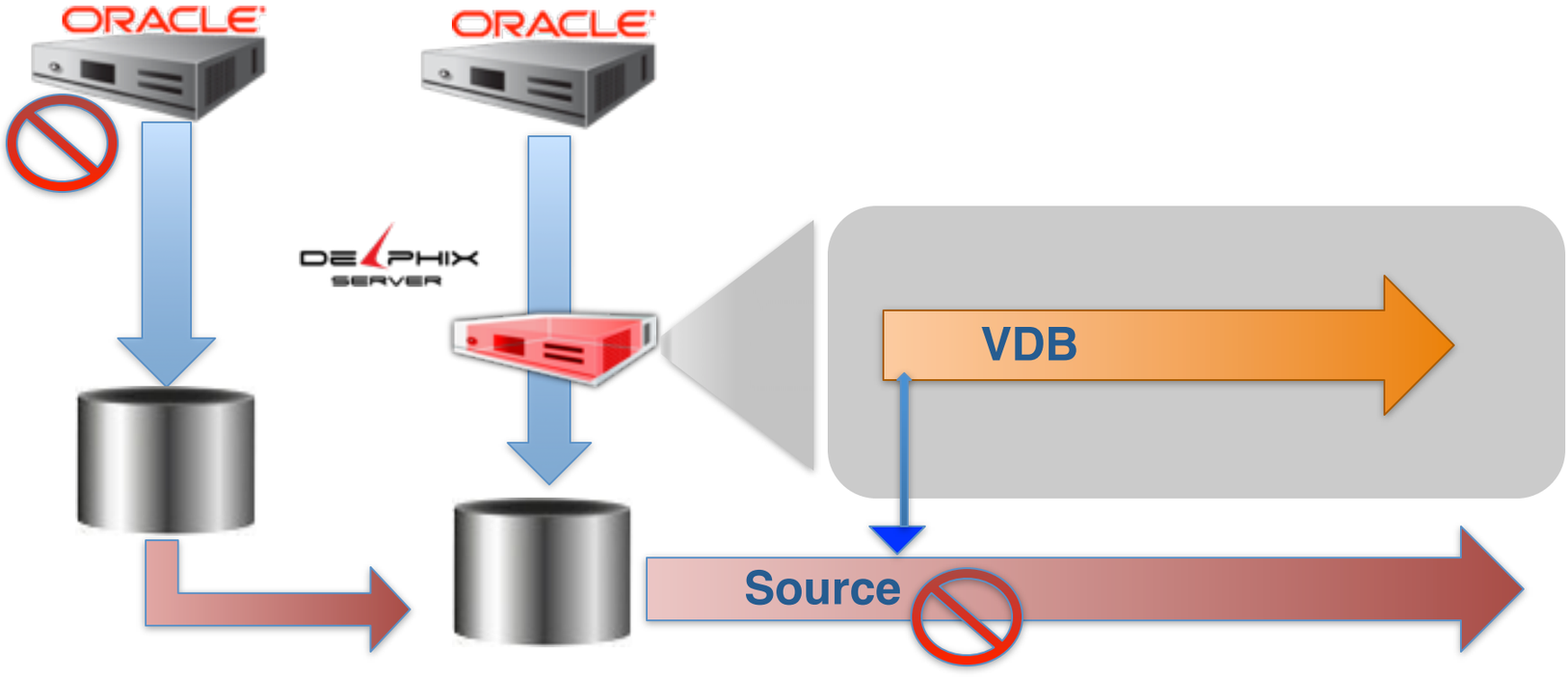
Fresh data

Full clones

Many Clones

Fast = Fresh
Full = Quality
Many = jet pack on development

Rollback / Time machine



DELPHIX®

easy as 3 minutes and 3 clicks

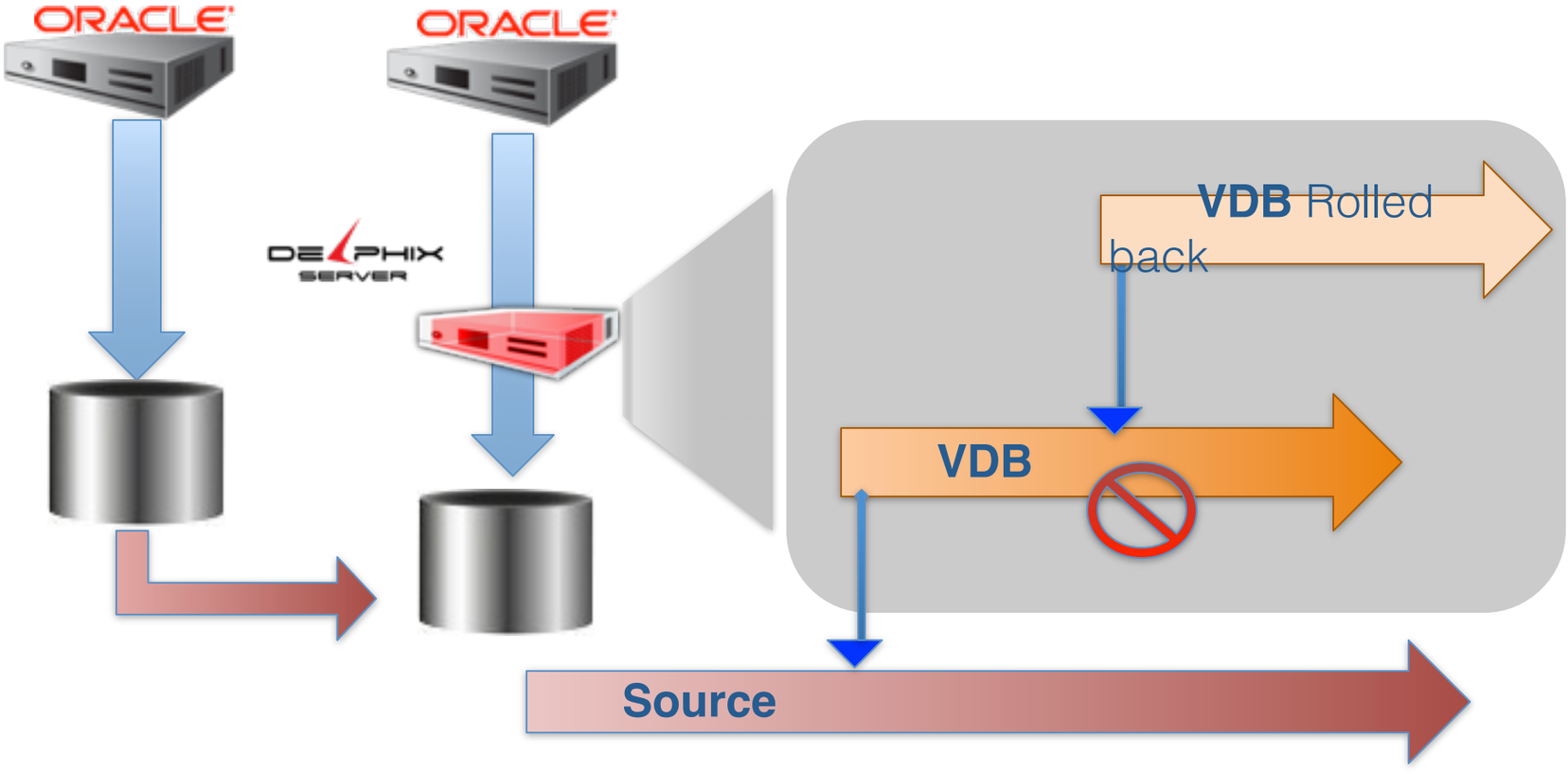
Com - DBA dropped movie titles table, 8 hour to restore a backup but only 2 minutes with Delphix
PG - operator entered Euro instead of US $
Fid - Oracle bug caused logical corruption on Dataguard, wouldn't start,

Time machine 3 minutes and 3 clicks

Eliminate Data Guard = 2 or 3 (even 4) dataguard
24 hours, 48 hours ago

dataguard for reports , no writes , breaks many reports
replace
Data domain
Avamar
Datavault

Rollback / Time machine



DELPHIX®

HD - developer truncated 232 Million row table, wanted it back

Cloning multiple source databases at the same time can be a daunting task

DELPHIX®

Like herding cats
One example with our customers is Informatica
Who had a project to integrate 6 databases into one central database
The time of the project was estimated at 12 months
With much of that coming from trying to orchestrating
Getting copies of the 6 databases at the same point in time

Informatica had a 12 month project to integrate 6 databases.
After installing Delphix they did it in 6 months.

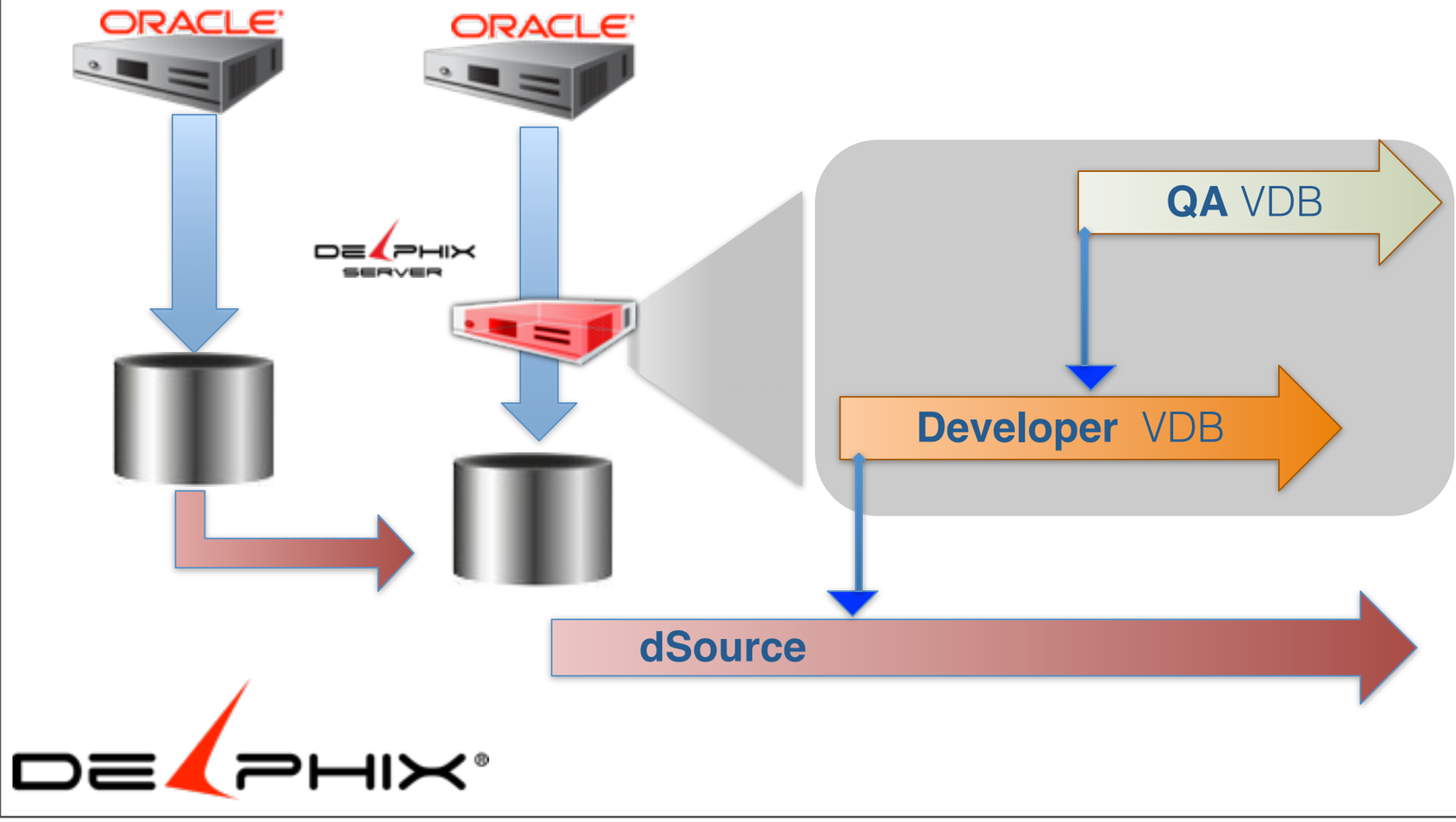
“I looked like a hero”
Tony Young, CIO
Informatica

Informatical had a 12 month project to integrate 6 databases.
After installing Delphix they did it in 6 months.

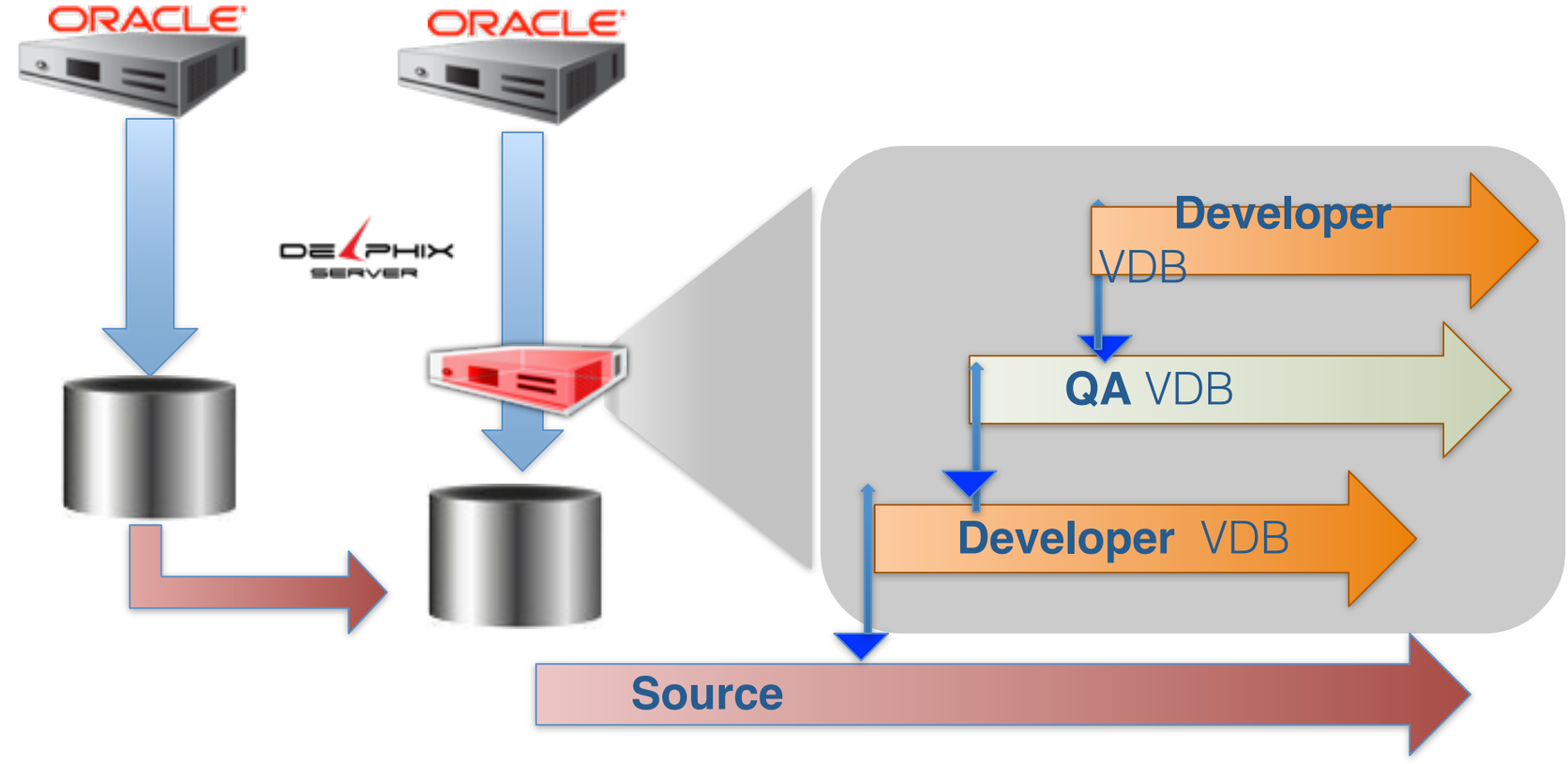
I delivered this early
I generated more revenue
I freed up money and put it into innovation

won an award with Ventana Research for this project

Pass to QA

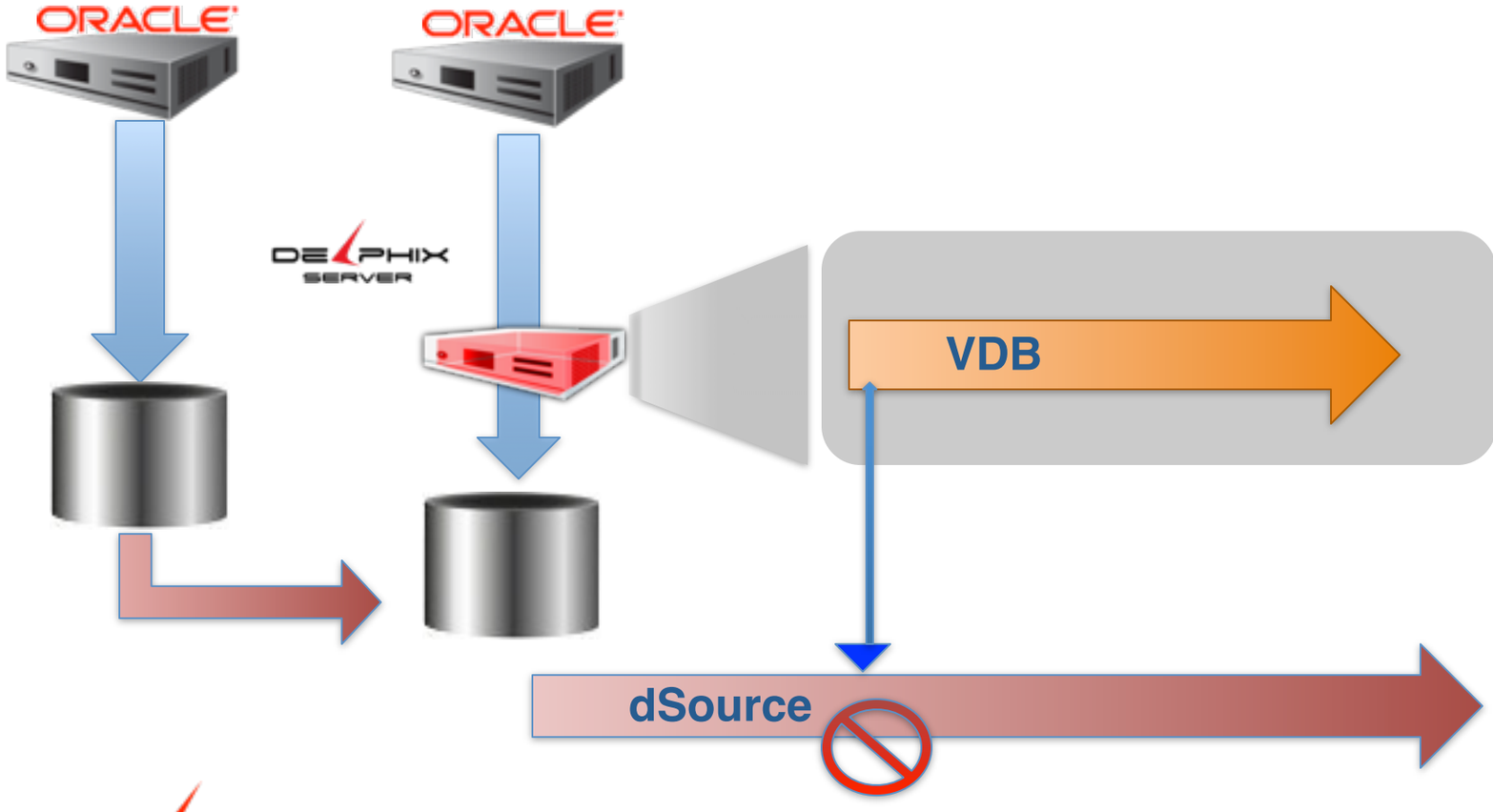

ORACLE
ORACLE
DELPHIX
SERVER
QA VDB
Developer VDB
dSource
DELPHIX

Punt to Dev



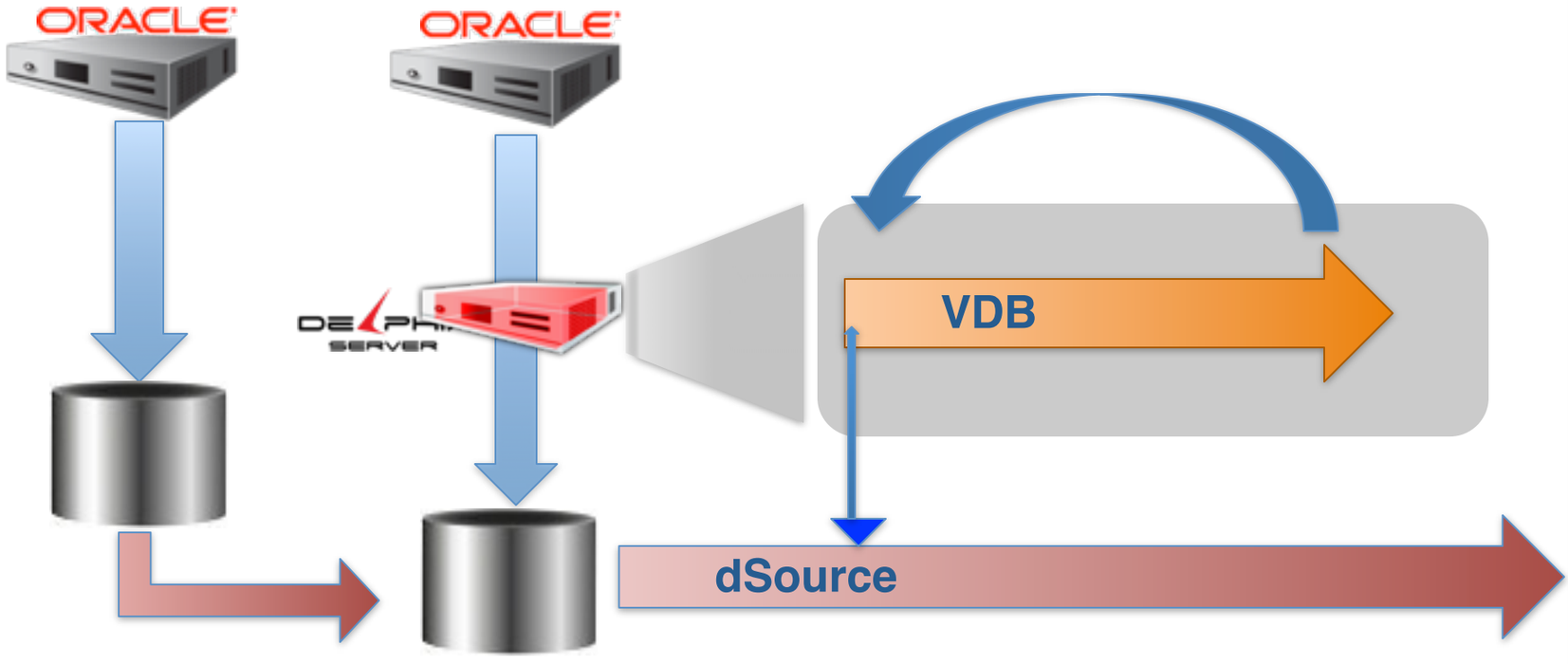
DELPHIX®

Forensic Analysis



DELPHIX®

Upgrades, Patches, RAT



DELPHIX®

50 days of backups in the size of one bac

50 days of backups in the size of one backup

Self Service

"perhaps the single largest storage consolidation opportunity history“

over 10 times

DELPHIX®

If every **MB** was an **Inch**

```
300,000 customers
       12 copies on average
       100 GB avg size
                           PB  TB  GB
       300000*12*100    = 360,000,000
       300000*1*.3*100  =   9,000,000
                          351 PB

        e   p   t   g
        1,191,290,000   feet to moon,
          132,000,000   feet around the earth
         e   p   t   g  m  k  b
        15,133,979,520 inches to the moon

     e   p   t   g  m  k  b
       351,000,000,000
        15,133,979,520 inches to the moon

     e  p  t  g  m  k  b
       351000000000
        15133979520 inches to the moon
```

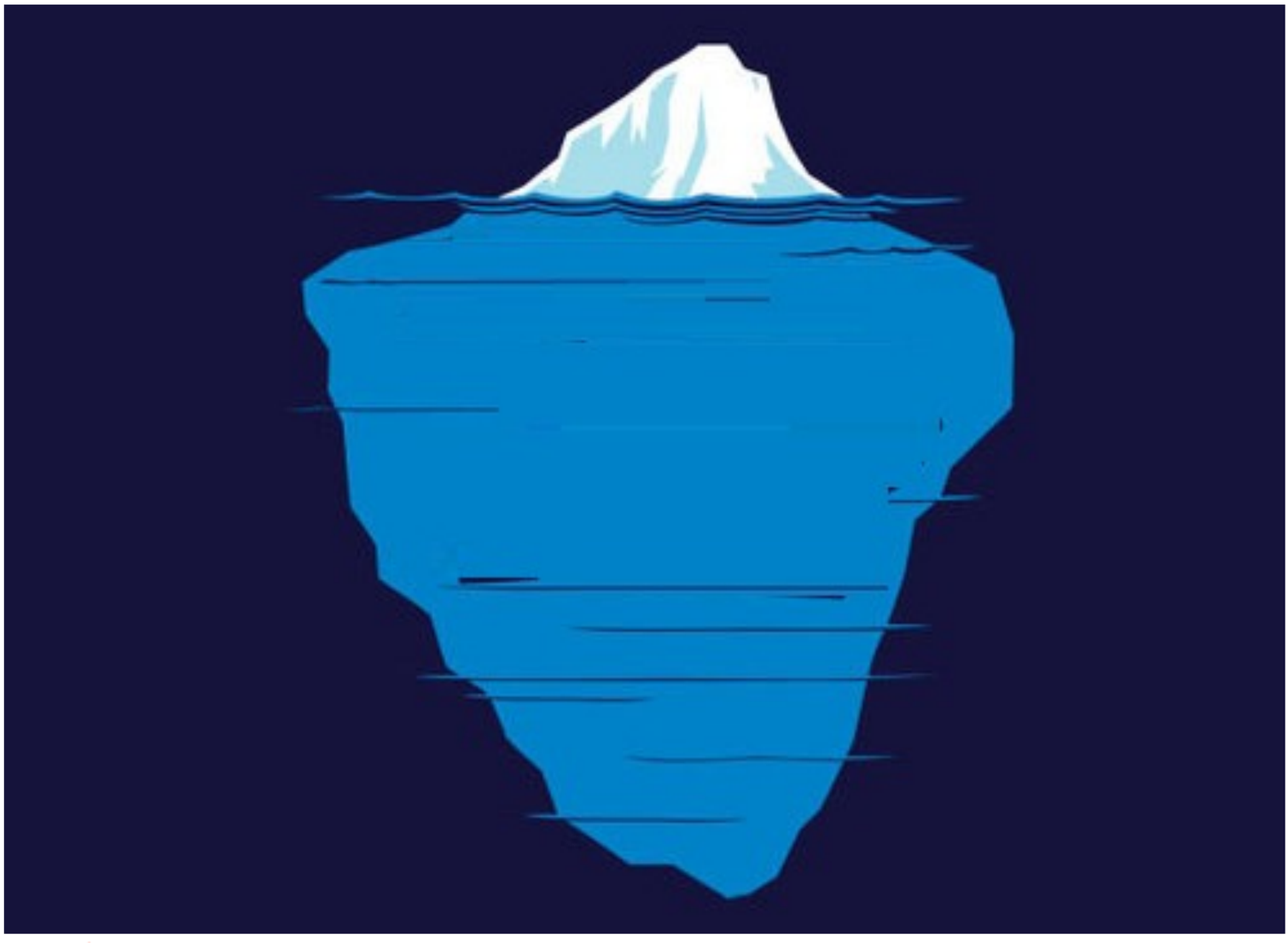
DELPHIX®

Storage

GP 480TB down to 60TB (1/8th)
SoCO 600TB down to 5TB
HD 720TB down to 8TB (create 19 x 36TB VDBs)

Time

PH 10 hours to 10 minutes
M's 4000 hours year down to 8 hours for refreshes
GP 21,000 man hours saved a year (refresh from 5 days to 5 minutes)
Stubhub freed up 2 full DBAs



DELPHIX®

Stubhub $5M estimated
GP $10M in hard cost savings
PG $12M a year estimates
DB $18M with current 25 databases (expects $200M with roll out to 700 databases)

Eliminating delays saves Developer time $1-3K/day

Informatica – finished 2x faster
Stubhub - 2 x as many releases a year
KLAT - 5 x as many projects

QA/Quality
Stubhub - 20% less bugs in production, found full table scan that would have been missed on subsets

Rather than suffering IT inefficiencies
By dragging behind enormous amounts of
infrastructure and bureaucracy
required to provide database copies

Using Delphix eliminates the drag and
provides power through
agile data management software and
database virtualization

Rather than suffering IT inefficiencies
By dragging behind enormous amounts of
infrastructure and bureaucracy
required to provide database copies

Using Delphix eliminates the drag and
provides power through
agile data management software and
database virtualization

A database refresh in 15 minutes?
That is mind blowing!
Delphix nailed it for us.

- Matt Lawrence , Sr Director Wind River (Intel)

We **need Delphix to scale** our agile environment

– Tim Campos, CIO, Facebook

Summary

Like having a genie
Instantly gives you
full copies for free

Any OS
Any hardware
Any storage

Reduce storage
Free up DBA resources
Put a jet pack on projects

Fully Automated
2 minute full clones
End user provisioning

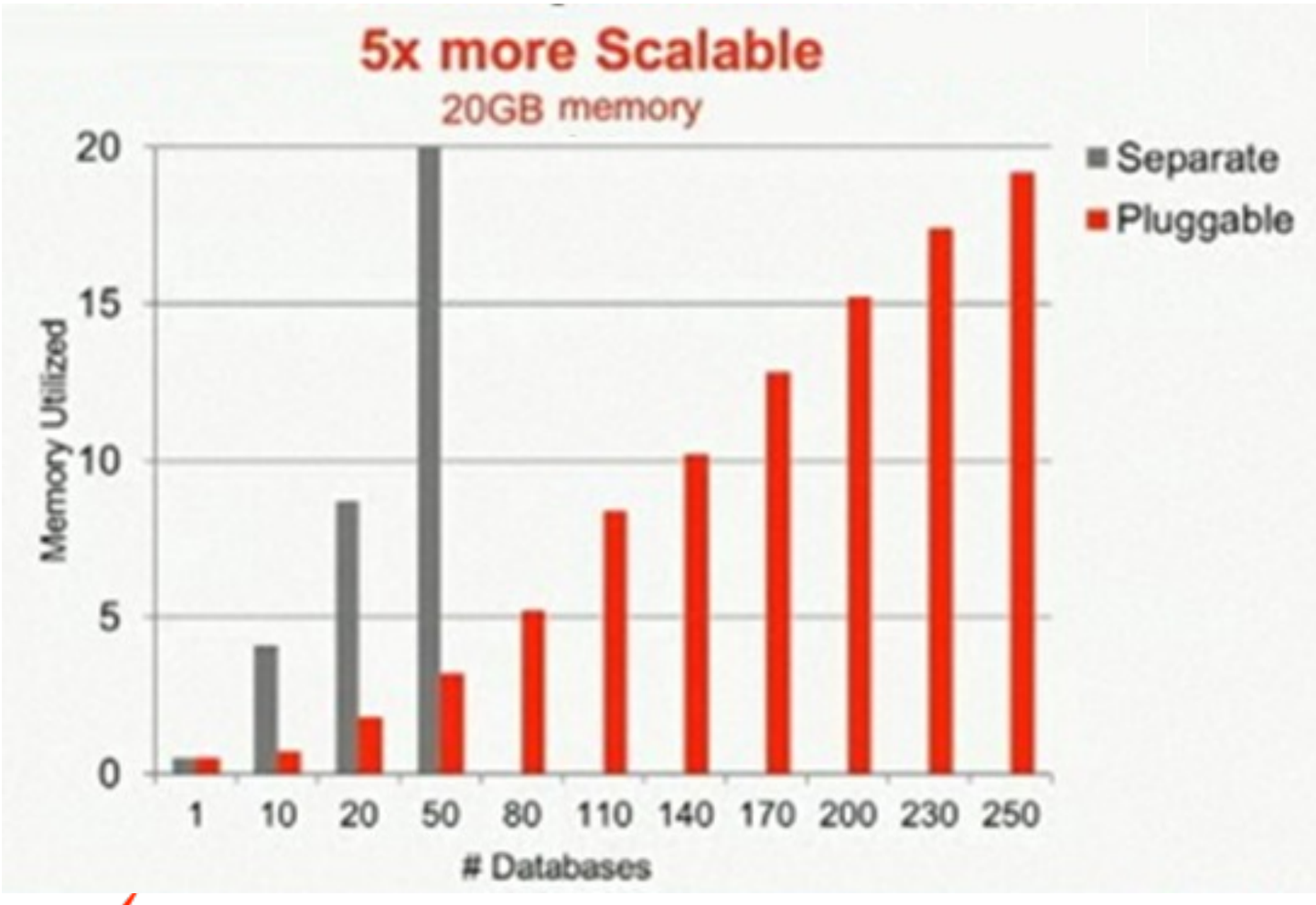
Fast query is the one you don't have to run
Fastest clone is the one you don't have to move
Eliminating data movement is the fasts easiest way to clone databases

Full automated instant thin provisioning of database clones
No one else can do it
though some provide the building blocks
but the building blocks have been around for 15 years

Conclusion : Enterprise

- ~~VMware Data Director~~
- EMC Timefinder
  - 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

Oracle 12c

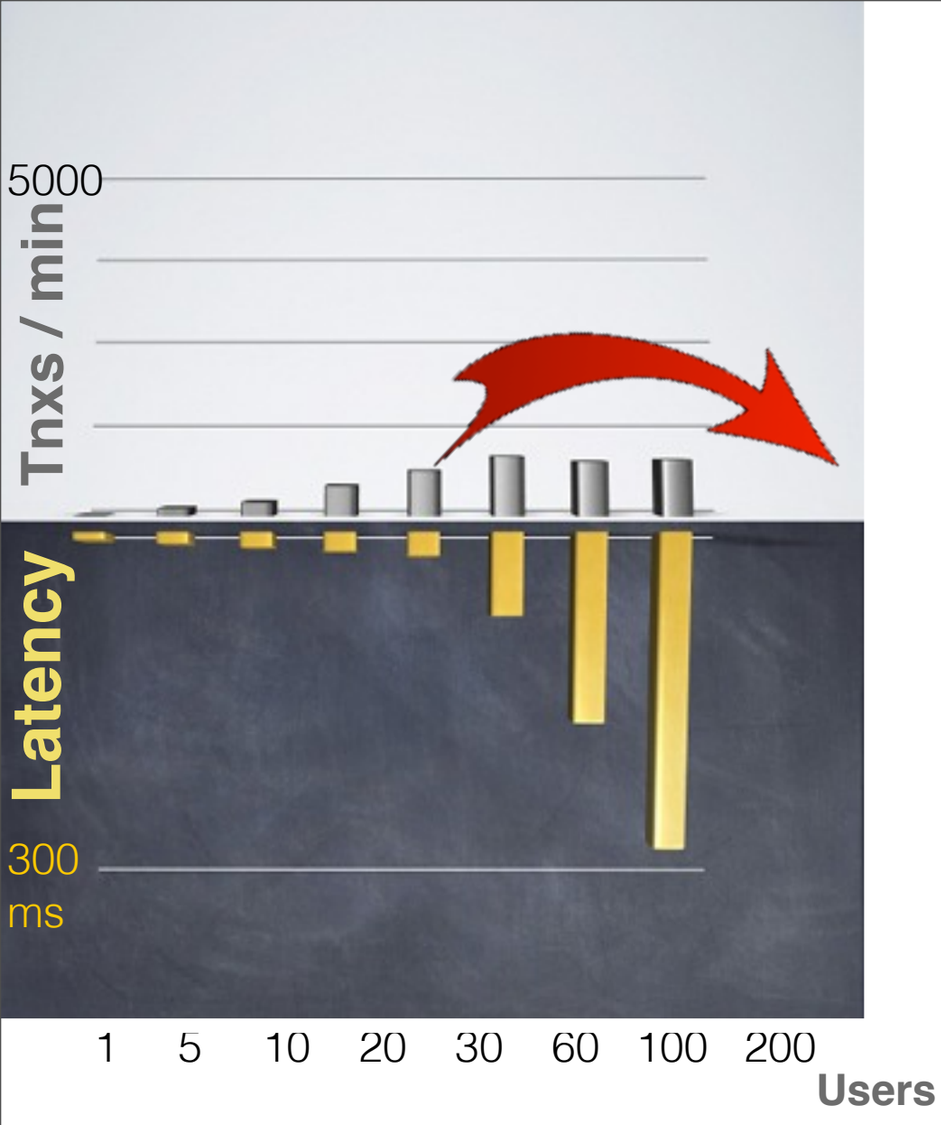


DELPHIX®

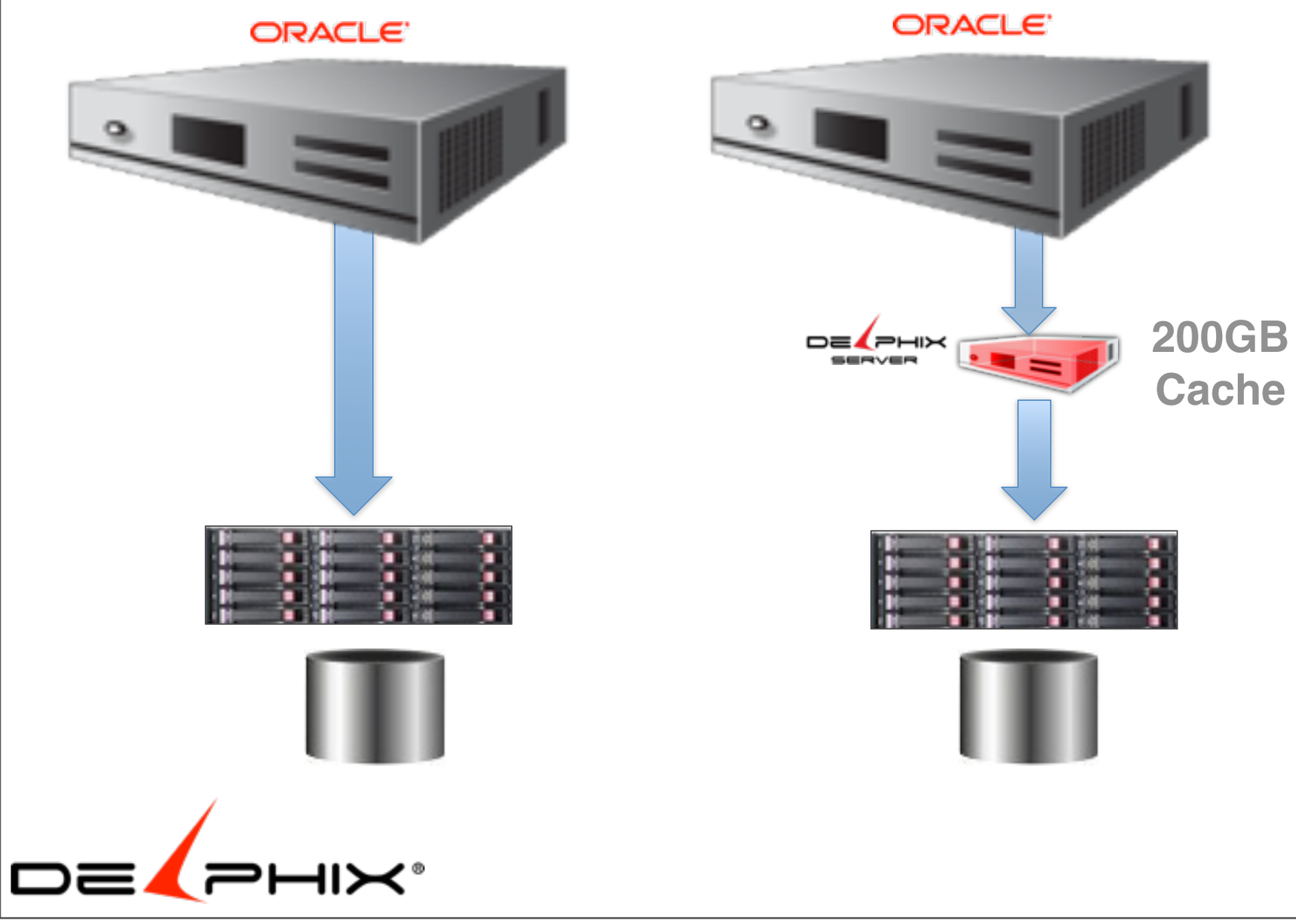
Once Last Thing

http://www.dadbm.com/wp-content/uploads/2013/01/12c_pluggable_database_vs_separate_database.png

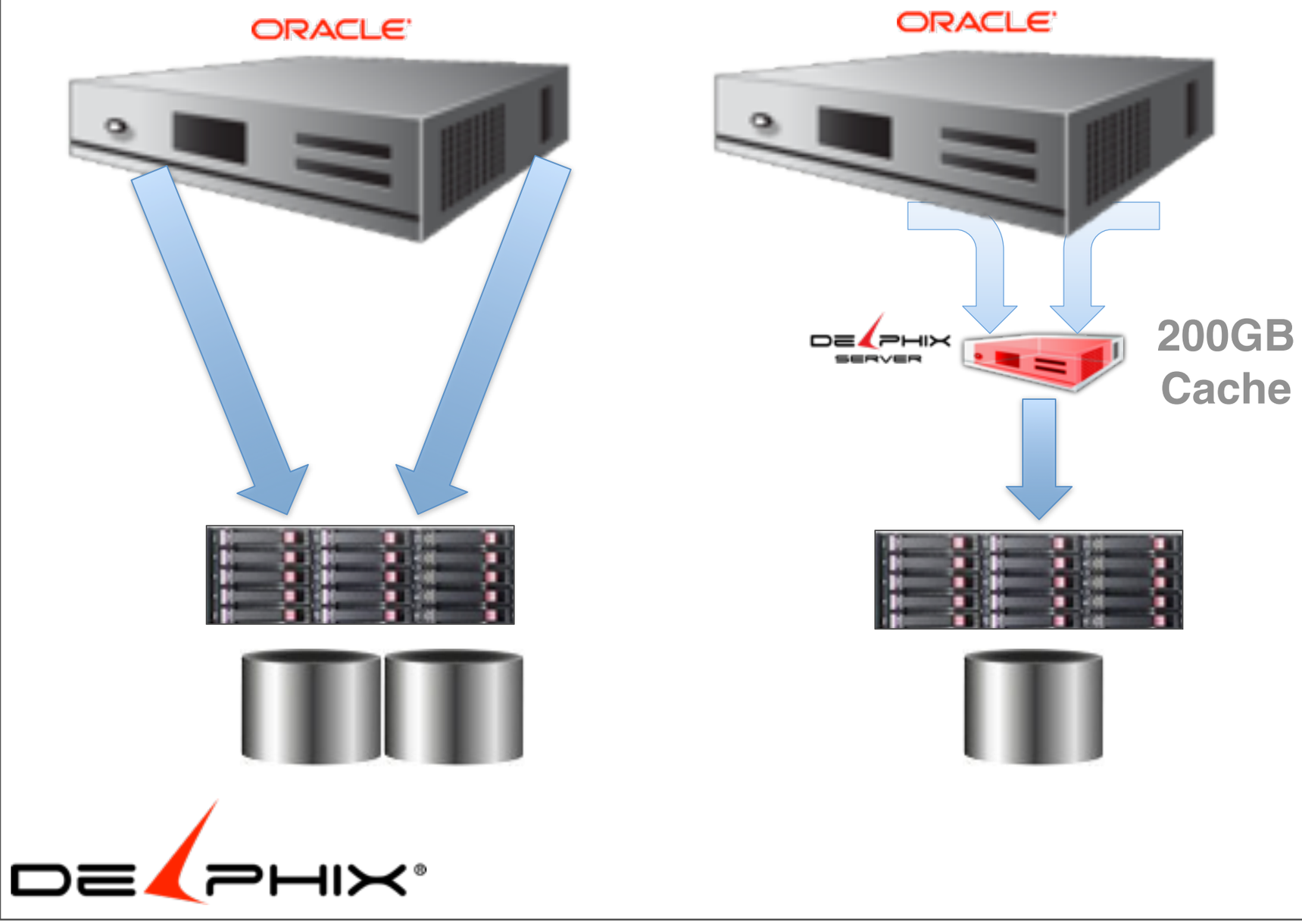
80MB buffer cache ?



DELPHIX®

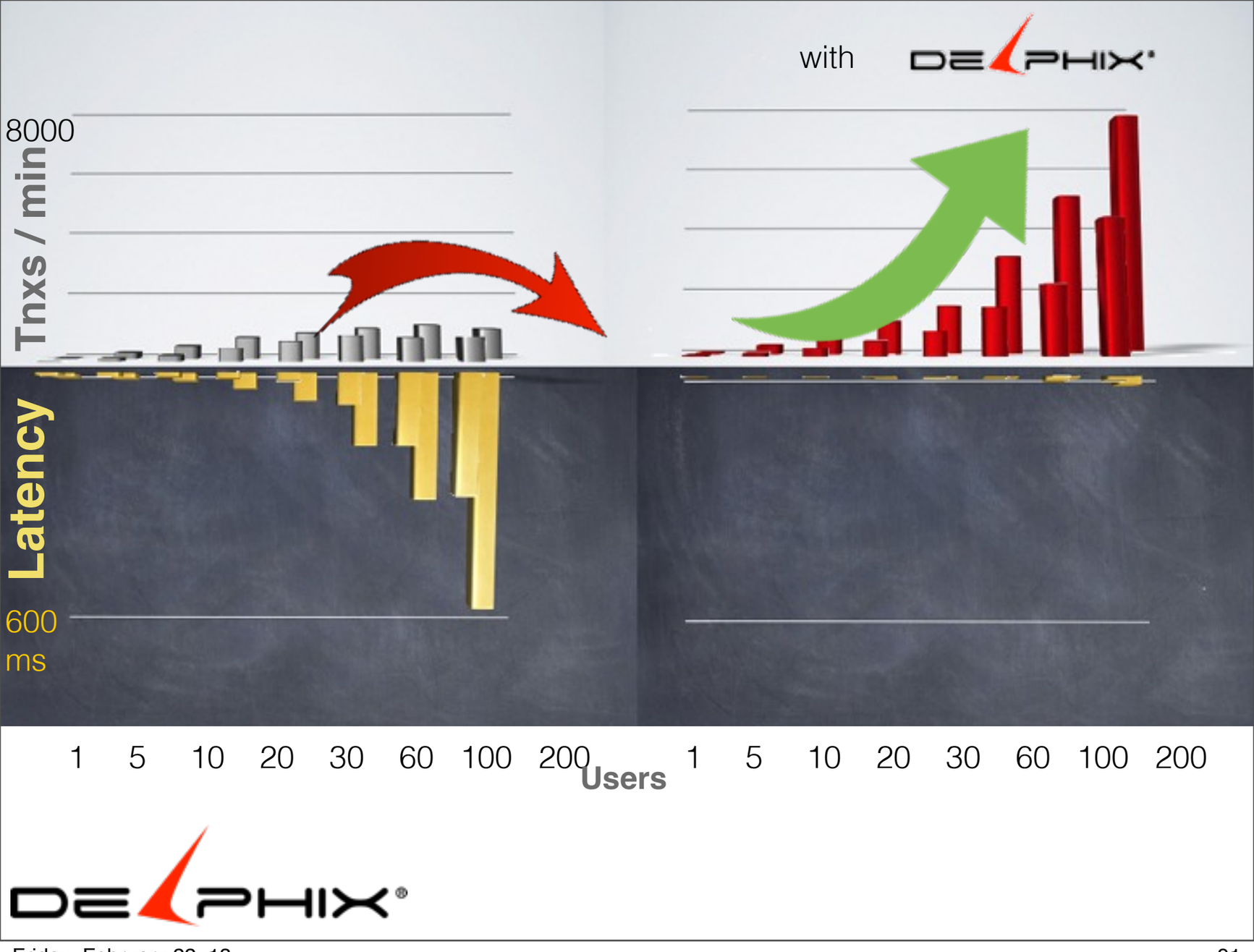

ORACLE
ORACLE
DELPHIX SERVER
200GB Cache
DELPHIX


with
5000
Tnxs / min
Latency
300
ms
1 5 10 20 30 60 100 200
Users
1 5 10 20 30 60 100 200


ORACLE
ORACLE
DELPHIX SERVER
200GB Cache
DELPHIX


with DELPHIX
8000
Tnxs / min
Latency
600
ms
1 5 10 20 30 60 100 200
Users
1 5 10 20 30 60 100 200

DELPHIX®


with DELPHIX
8000
Tnxs / min
Latency
600
ms
1 5 10 20 30 60 100 200
Users
1 5 10 20 30 60 100 200

DELPHIX®

Matrix of features

	CloneDB	ZFS Appliance	Delphix	Data Director	NetApp	EMC
Time Flow	No	Yes	Yes	No	Yes	No
Hardware Agnostic	Yes	No	Yes	Yes	No	No
Snapshots	No	Unlimited	Unlimited	31	255	16 (96 read only)
Snapshots of snapshots	No	Unlimited	Unlimited	30	255	1
Automated Snapshots	No	No	Yes	No	Yes	No
Automated Provisioning	No	No	Yes	No	No	No
Any O/S	Yes	Yes	Yes	No x86 only	Yes	Yes
Max size	None	None	None	?	16-100TB	?

VMWARE

http://www.virtuallyghetto.com/2010/10/how-to-control-maximum-number-of-vmware.html
http://communities.vmware.com/thread/303689

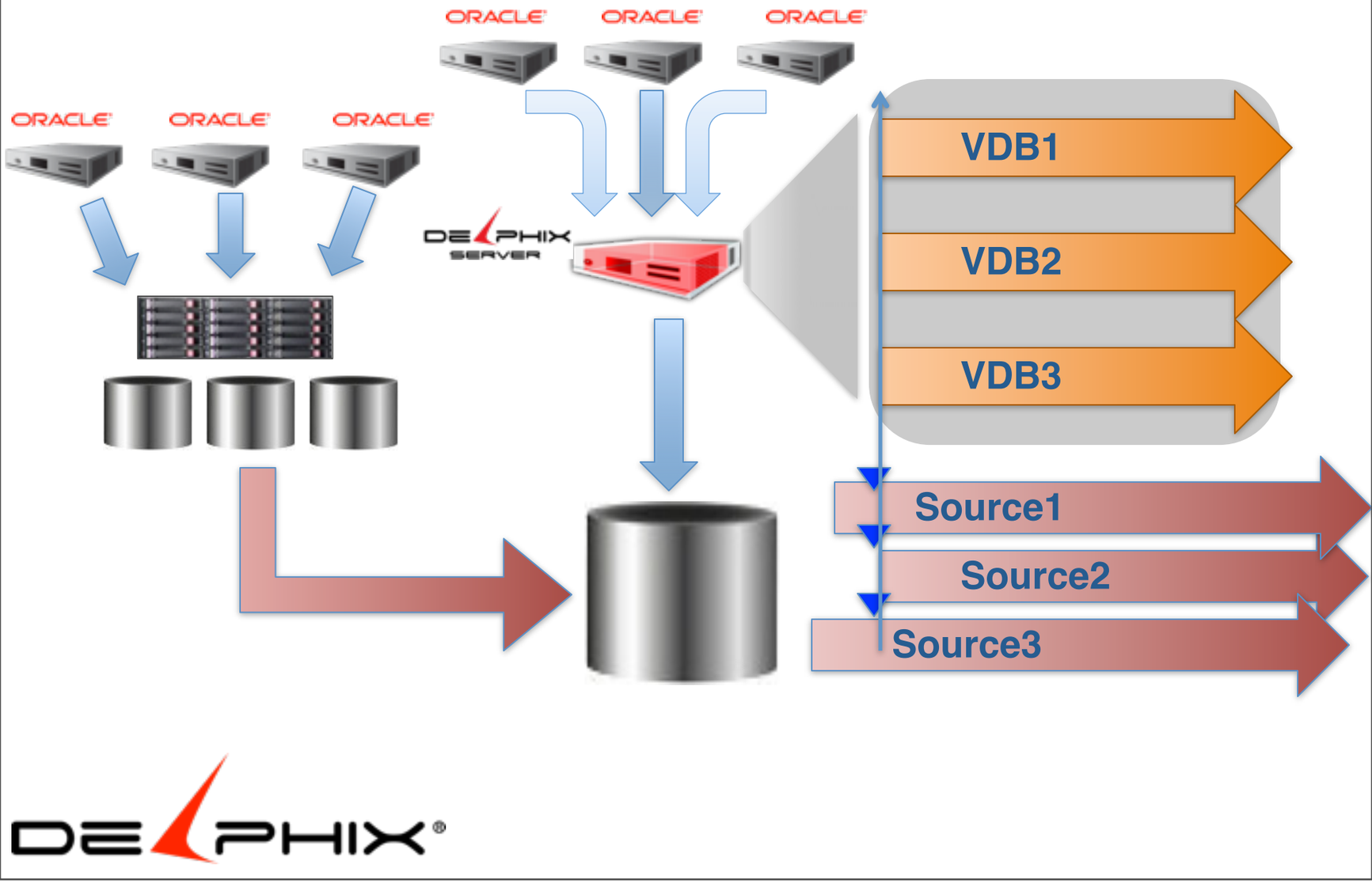
EMC
http://www.lascon.co.uk/snap-timefinder.php
https://community.emc.com/servlet/JiveServlet/previewBody/11789-102-1-45992/h8728-snapsure-oracle-dnfs-wp.pdf

Appendix

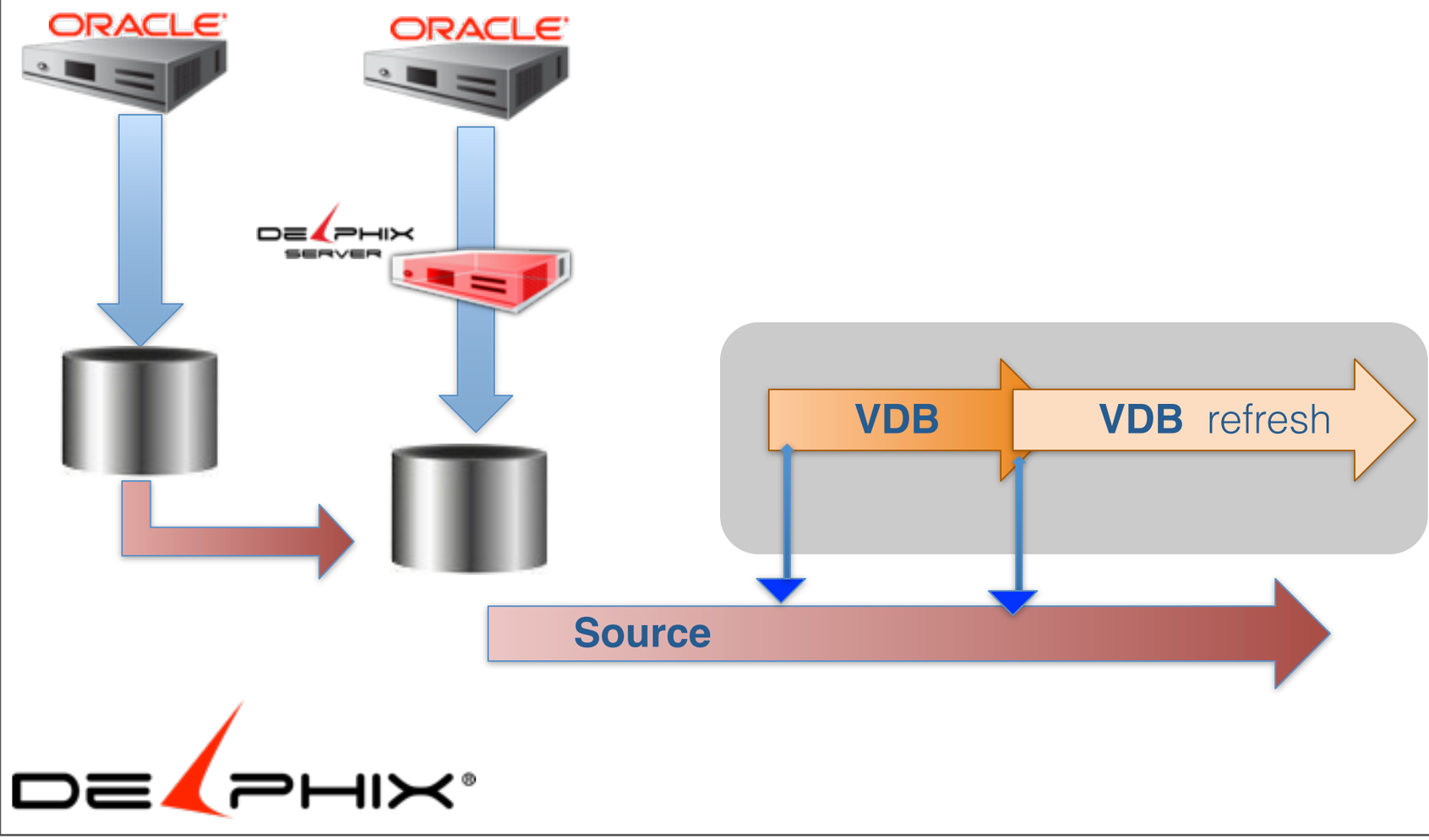
- CloneDB
  - http://www.oracle-base.com/articles/11g/clonedb-11gr2.php
- ZFS
  - http://hub.opensolaris.org/bin/download/Community+Group+zfs/docs/zfslast.pdf
- ZFS Appliance
  - http://www.oracle.com/technetwork/articles/systems-hardware-architecture/cloning-solution-353626.pdf
- Data Director
  - http://www.virtuallyghetto.com/2012/04/scripts-to-extract-vcloud-director.html
  - http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=1015180
- EMC
  - https://community.emc.com/servlet/JiveServlet/previewBody/11789-102-1-45992/h8728-snapsure-oracle-dnfs-wp.pdf
- NetApp
  - RAC provision example http://blog.flimatech.com/2008/02/07/how-to-create-a-

Other Use Cases

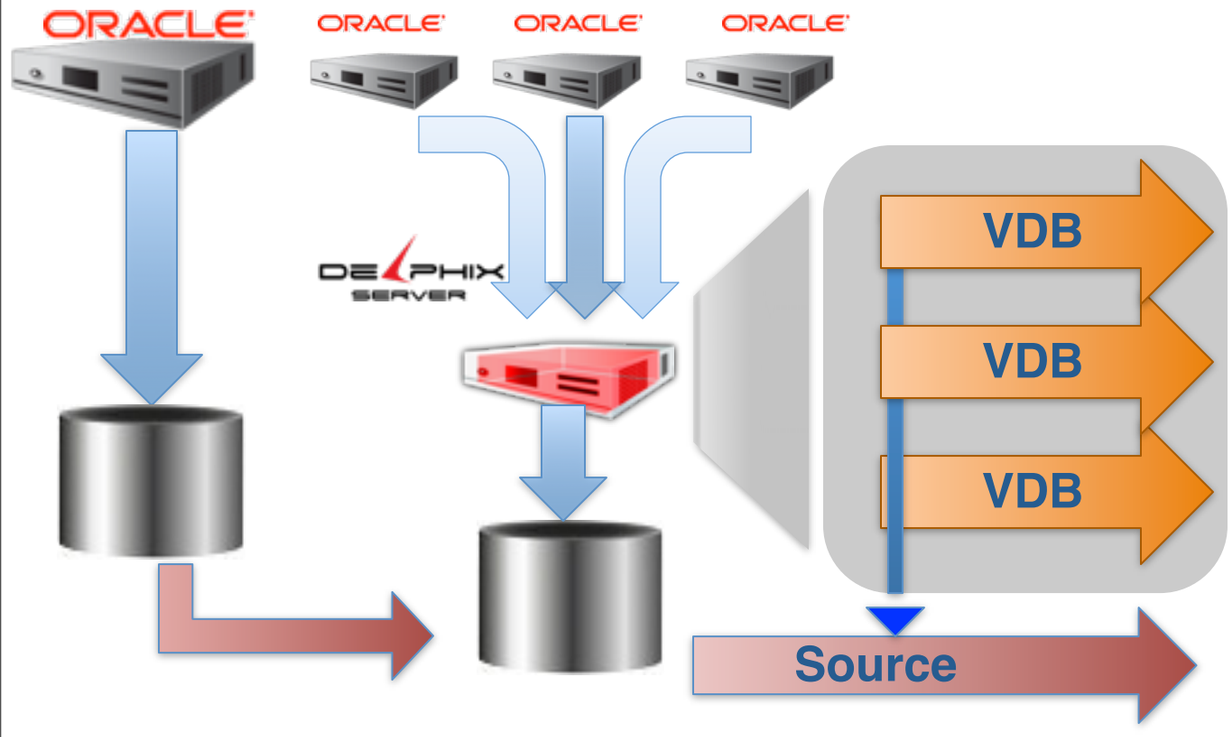
Collaborative sources



DELPHIX®

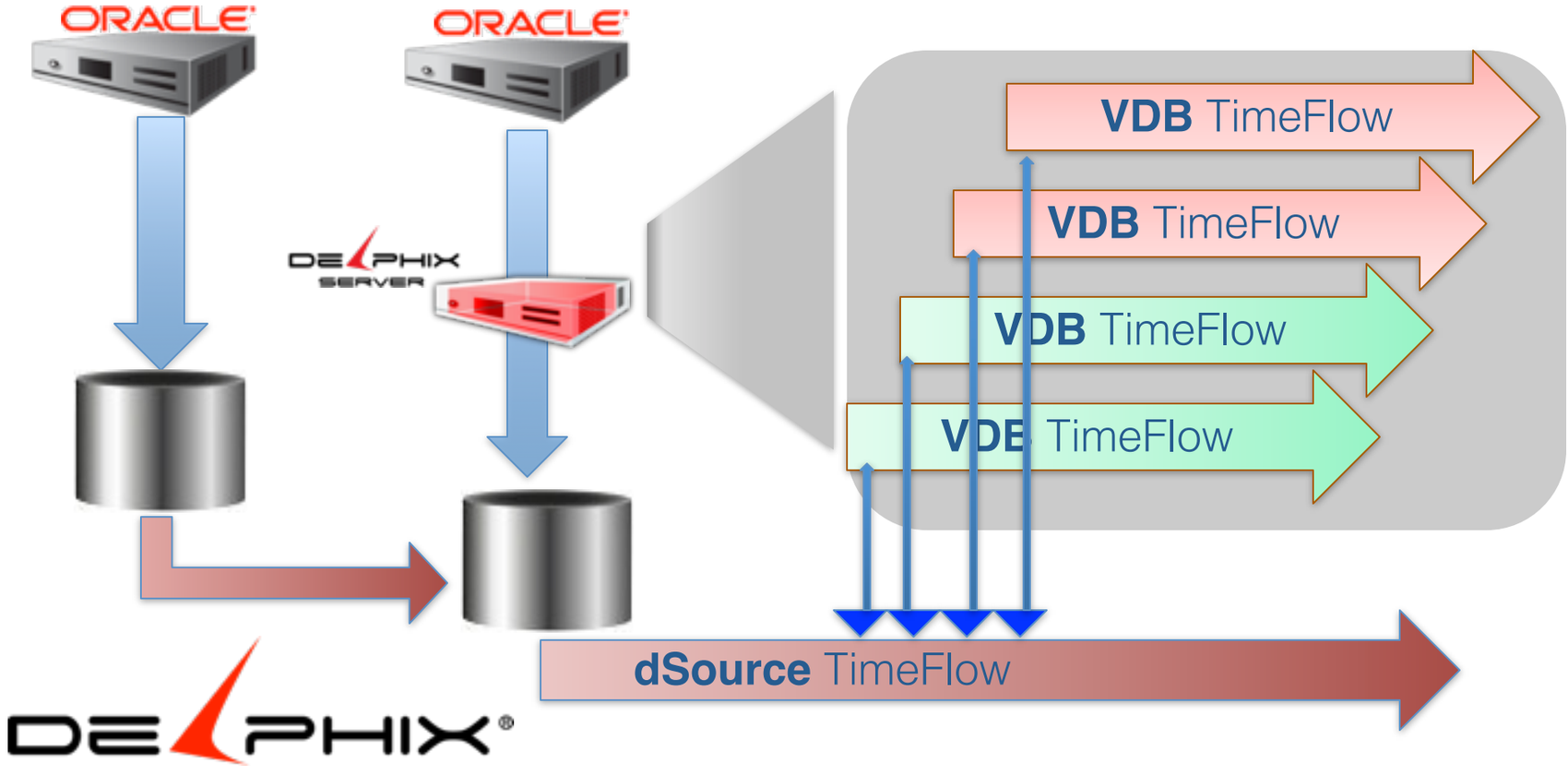


DELPHIX®

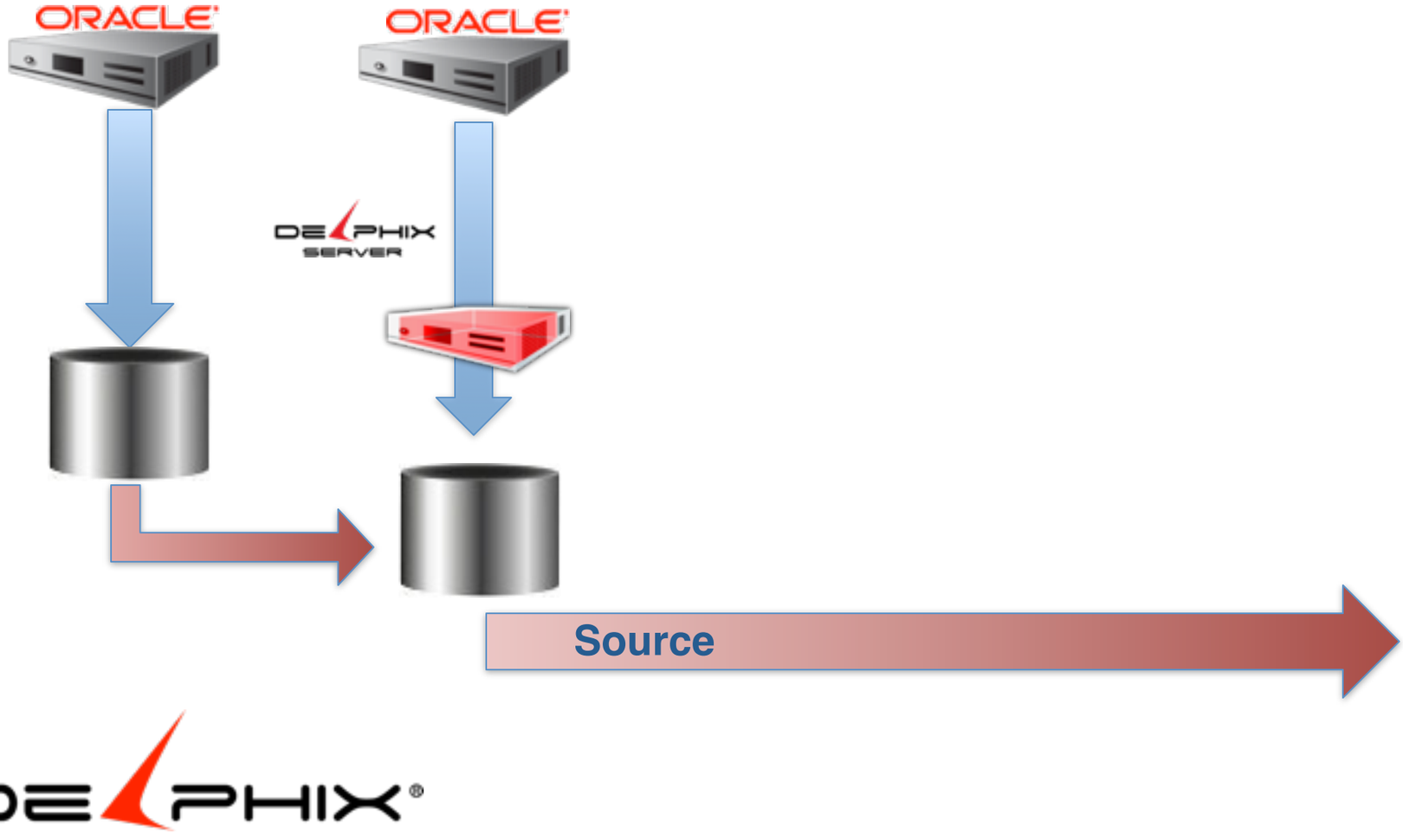

ORACLE
ORACLE
ORACLE
ORACLE
DELPHIX
SERVER
VDB
VDB
VDB
Source

DELPHIX®

Logical Corruption

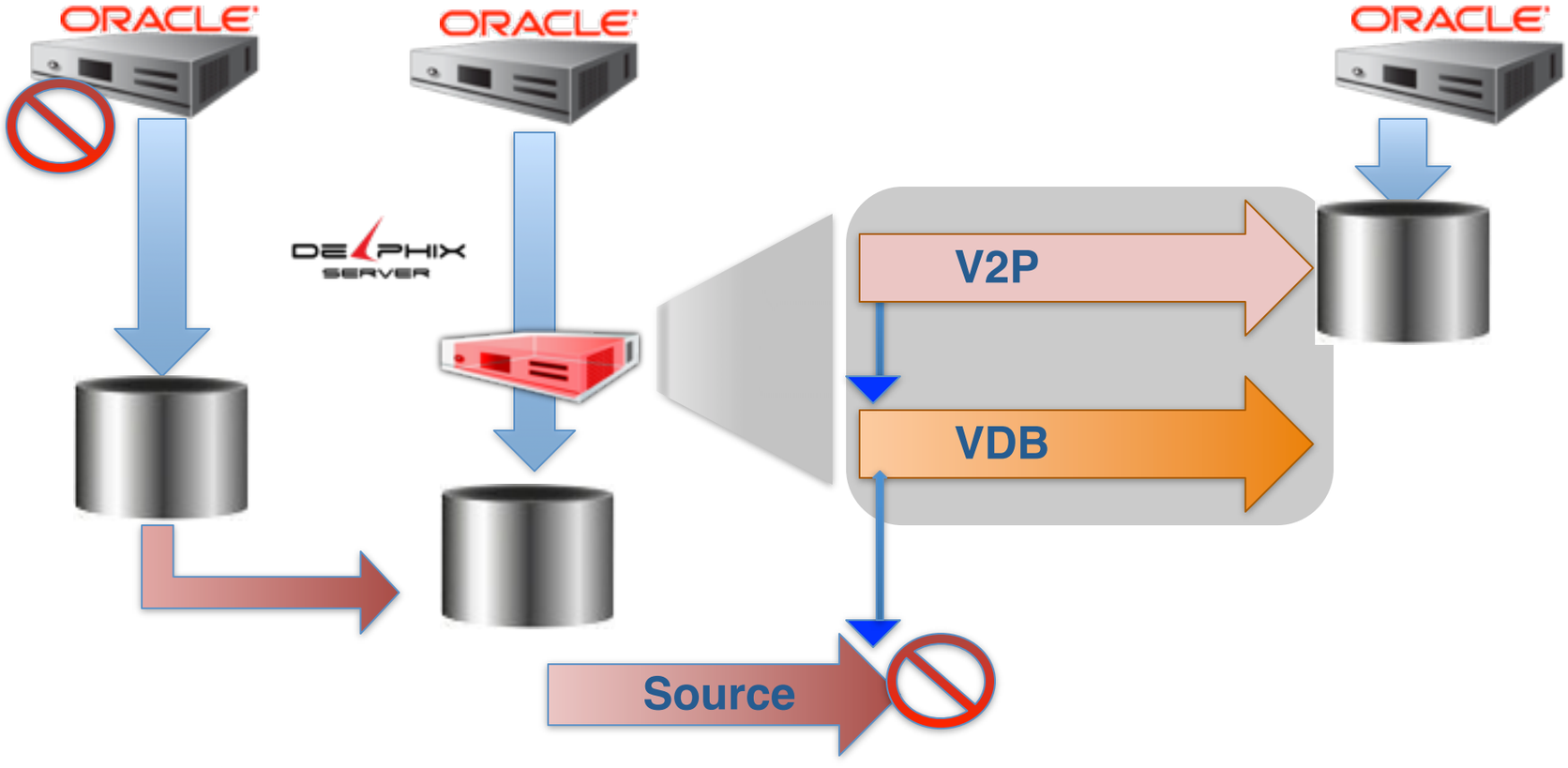


50 days for 1



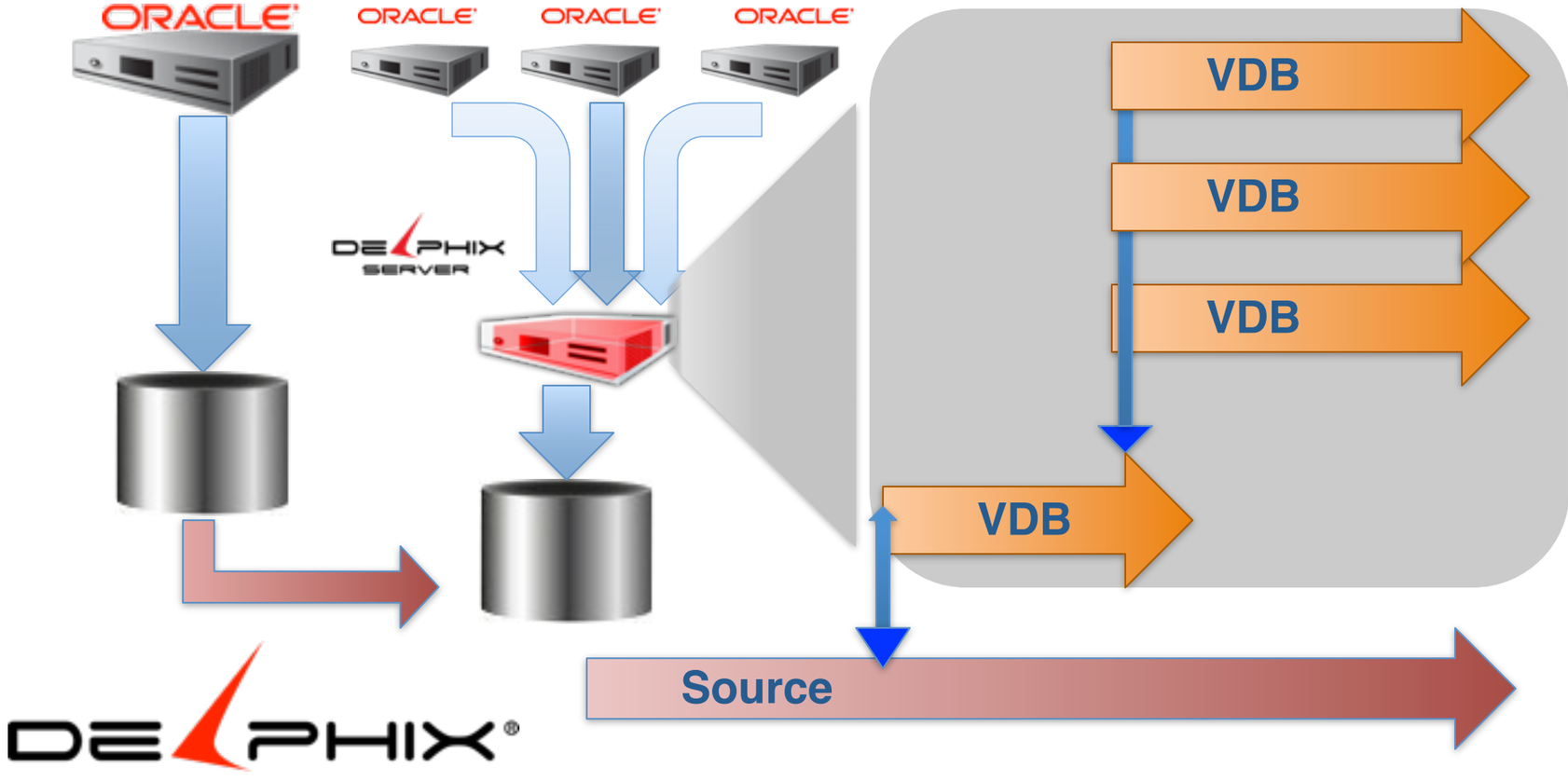
DELPHIX®

Recovery

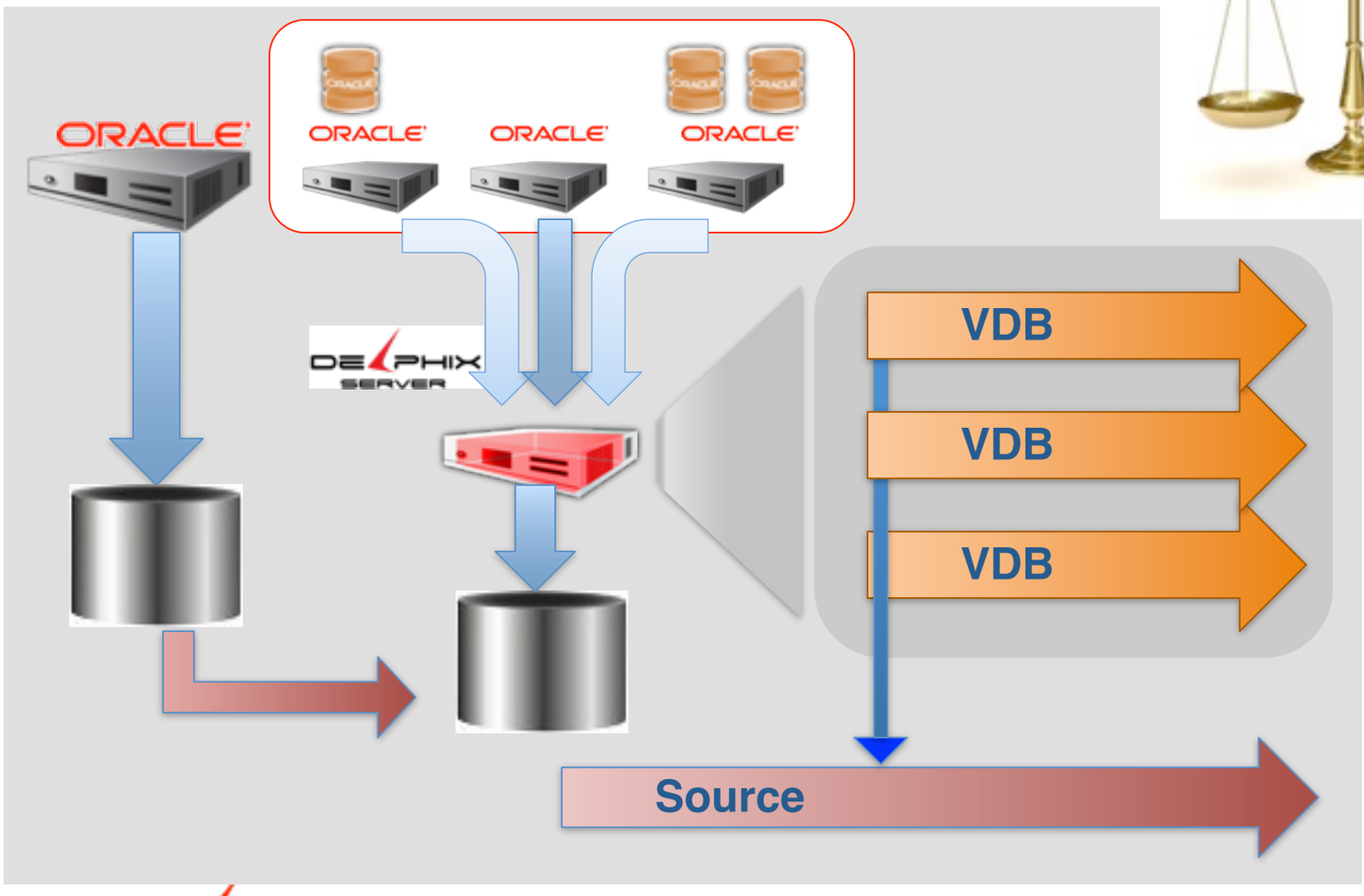


DELPHIX®

Baseline Masking

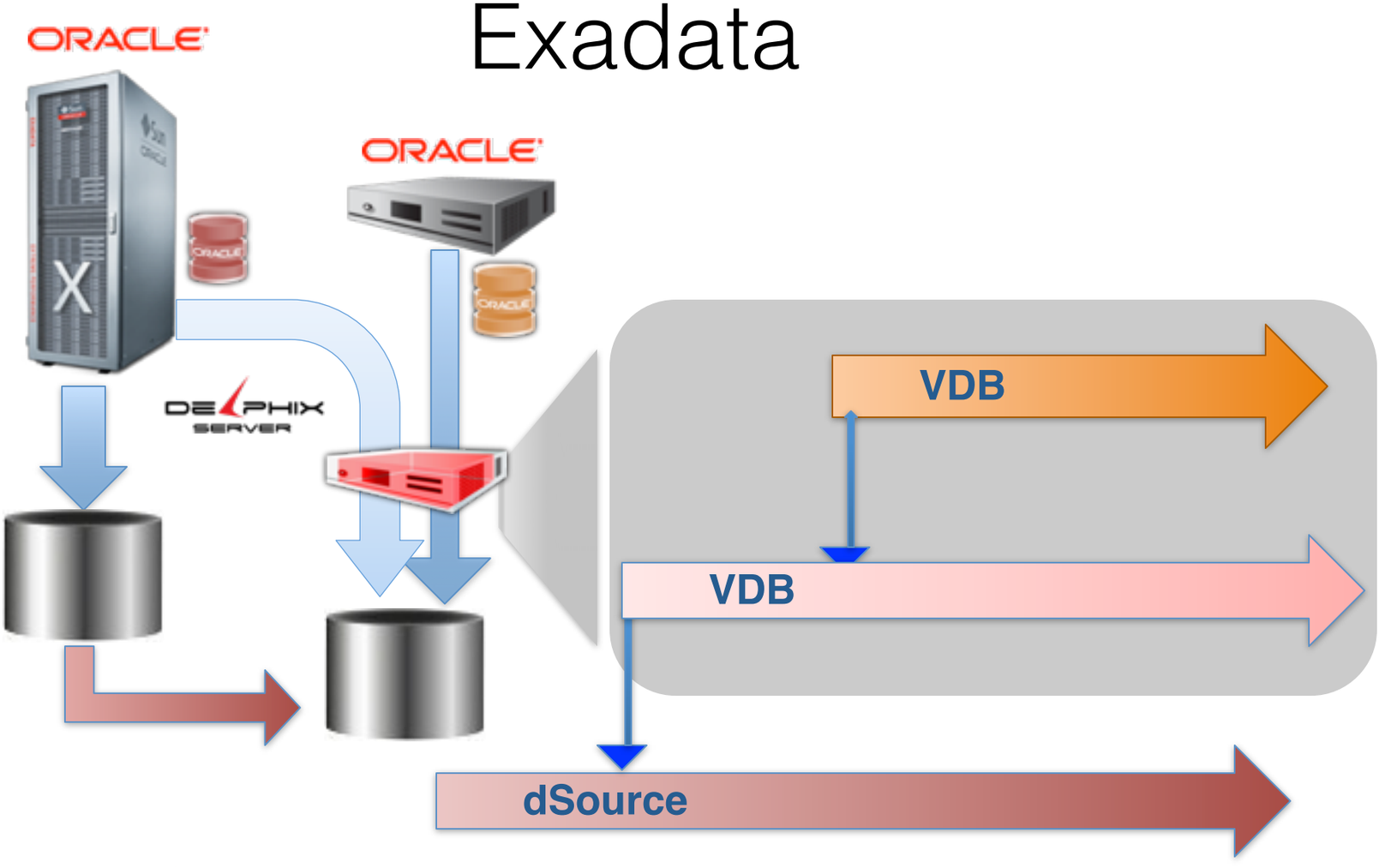


Load Balancing



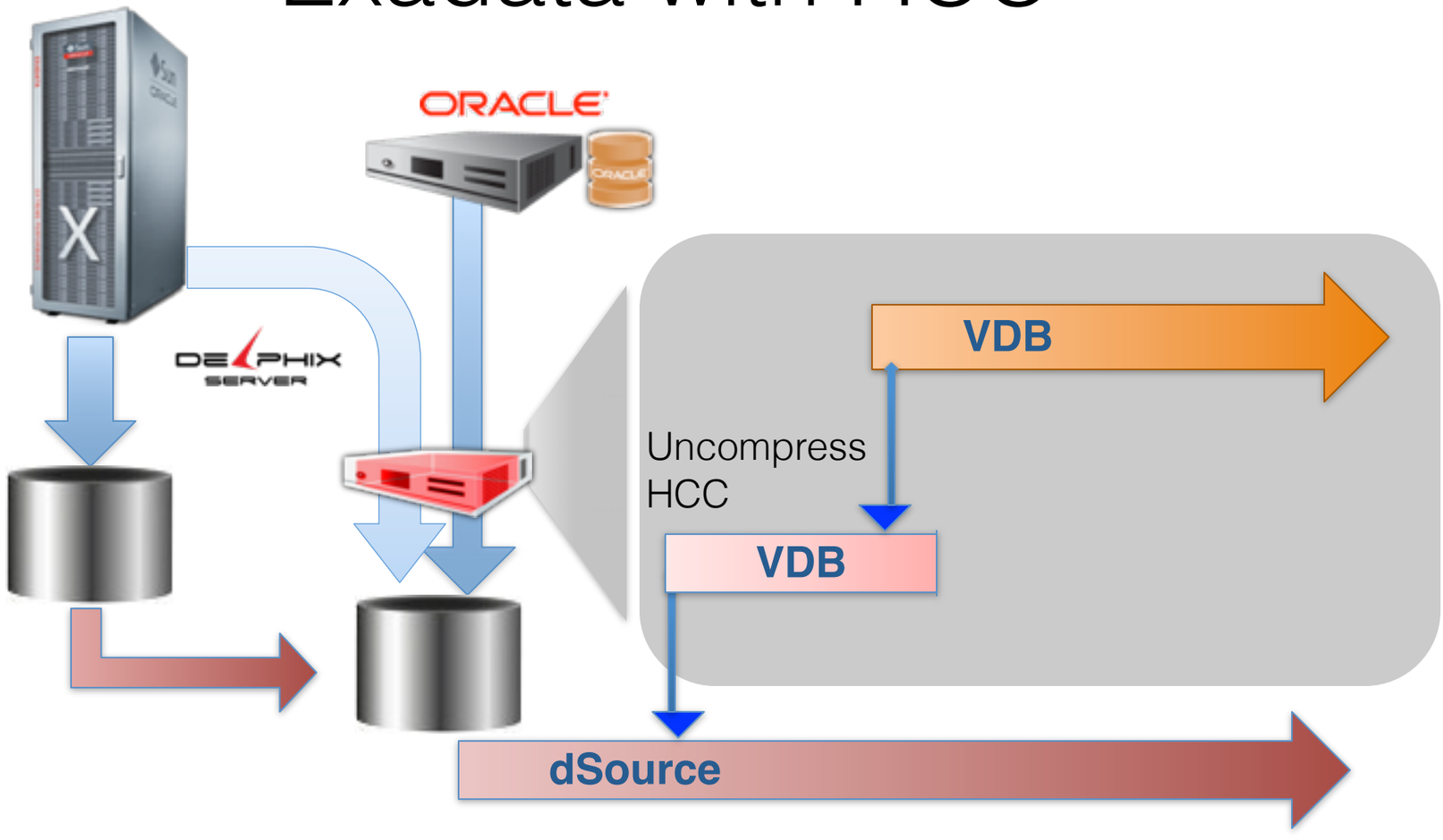
DELPHIX®

Exadata



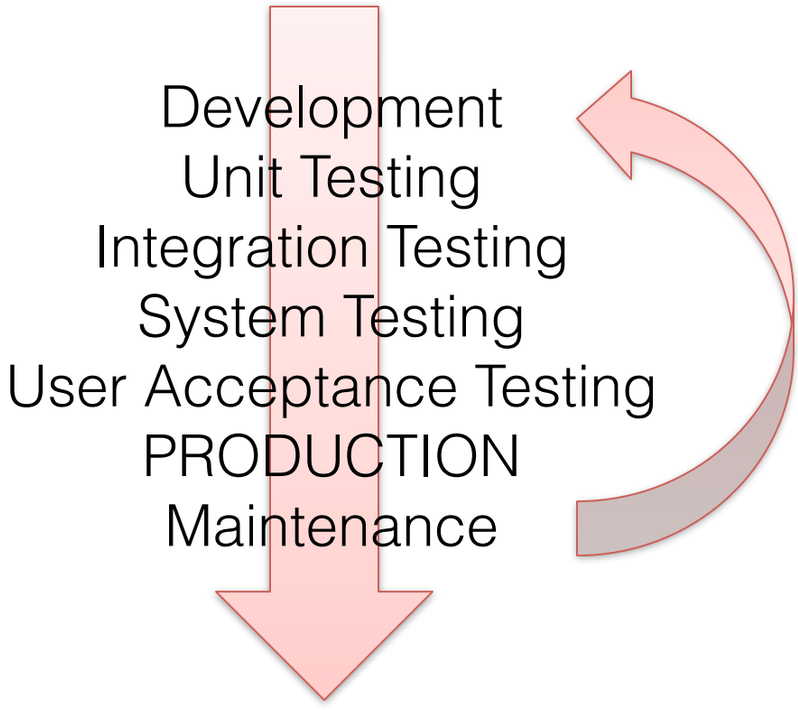
DELPHIX®

Exadata with HCC



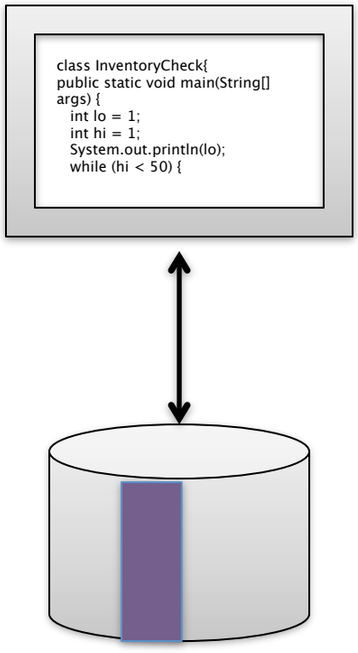
DELPHIX®

Common Stages

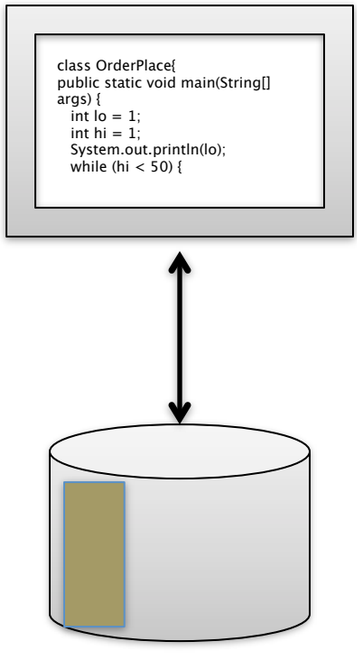


Development

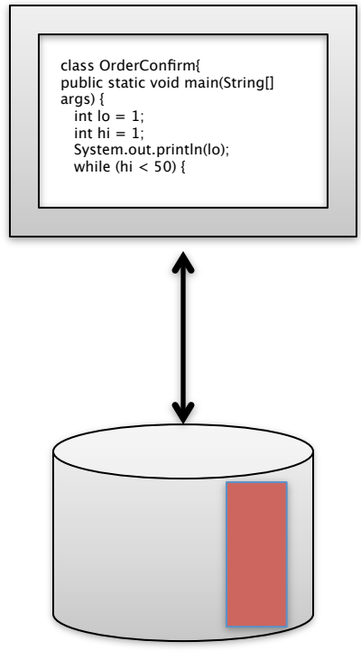
Module: Check Inventory



Module: Place Order

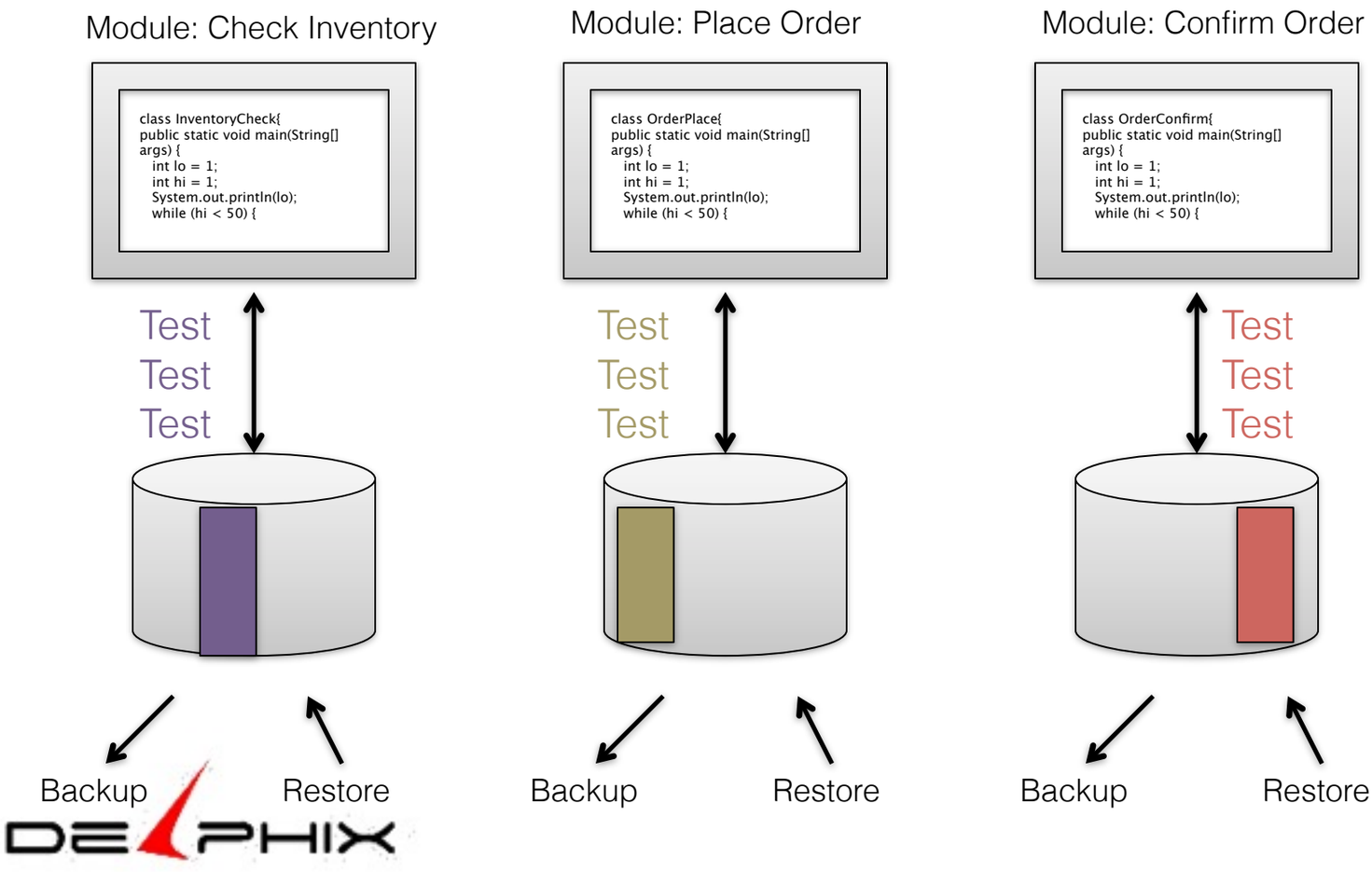


Module: Confirm Order

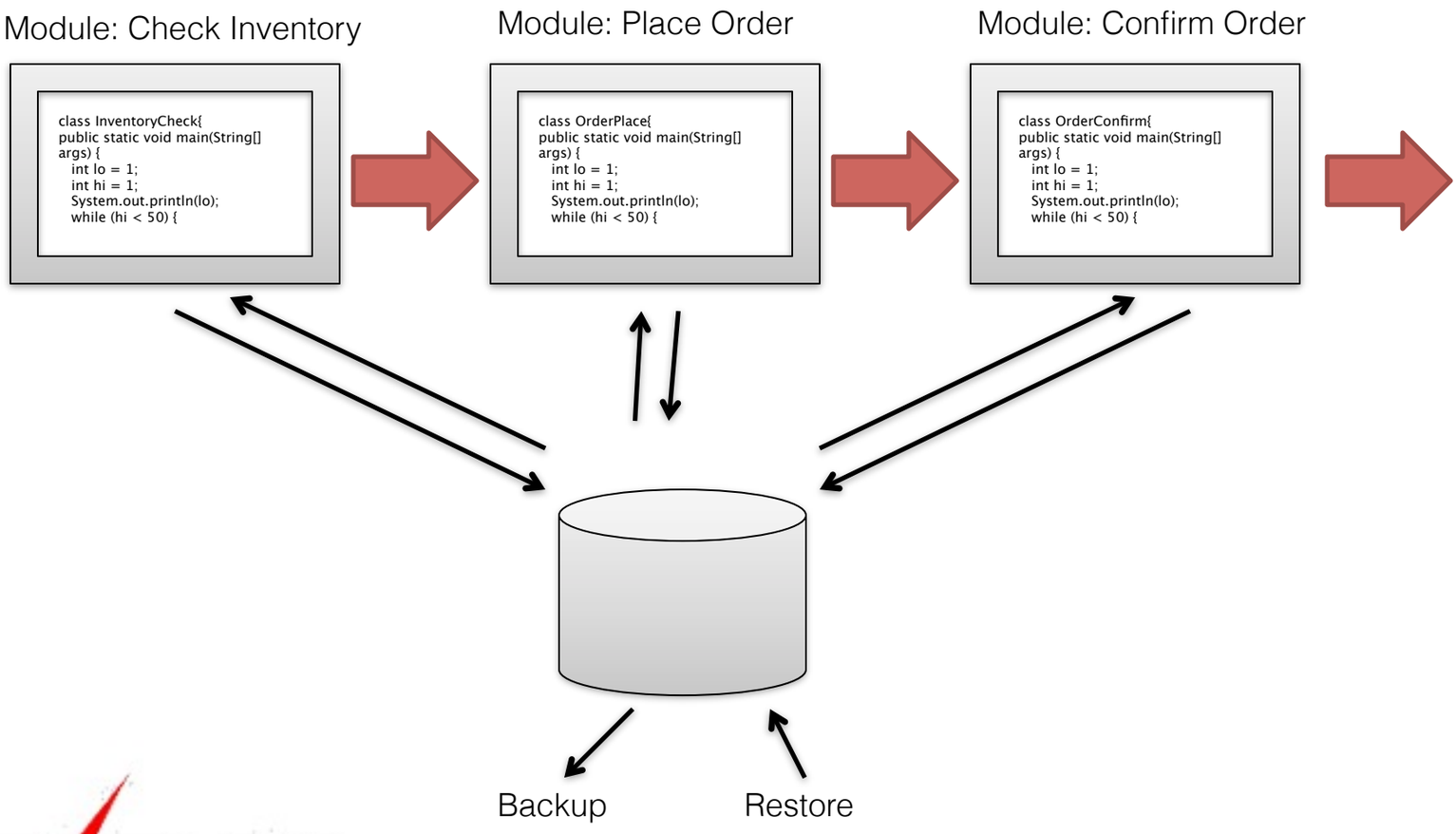


DELPHIX

Unit Test

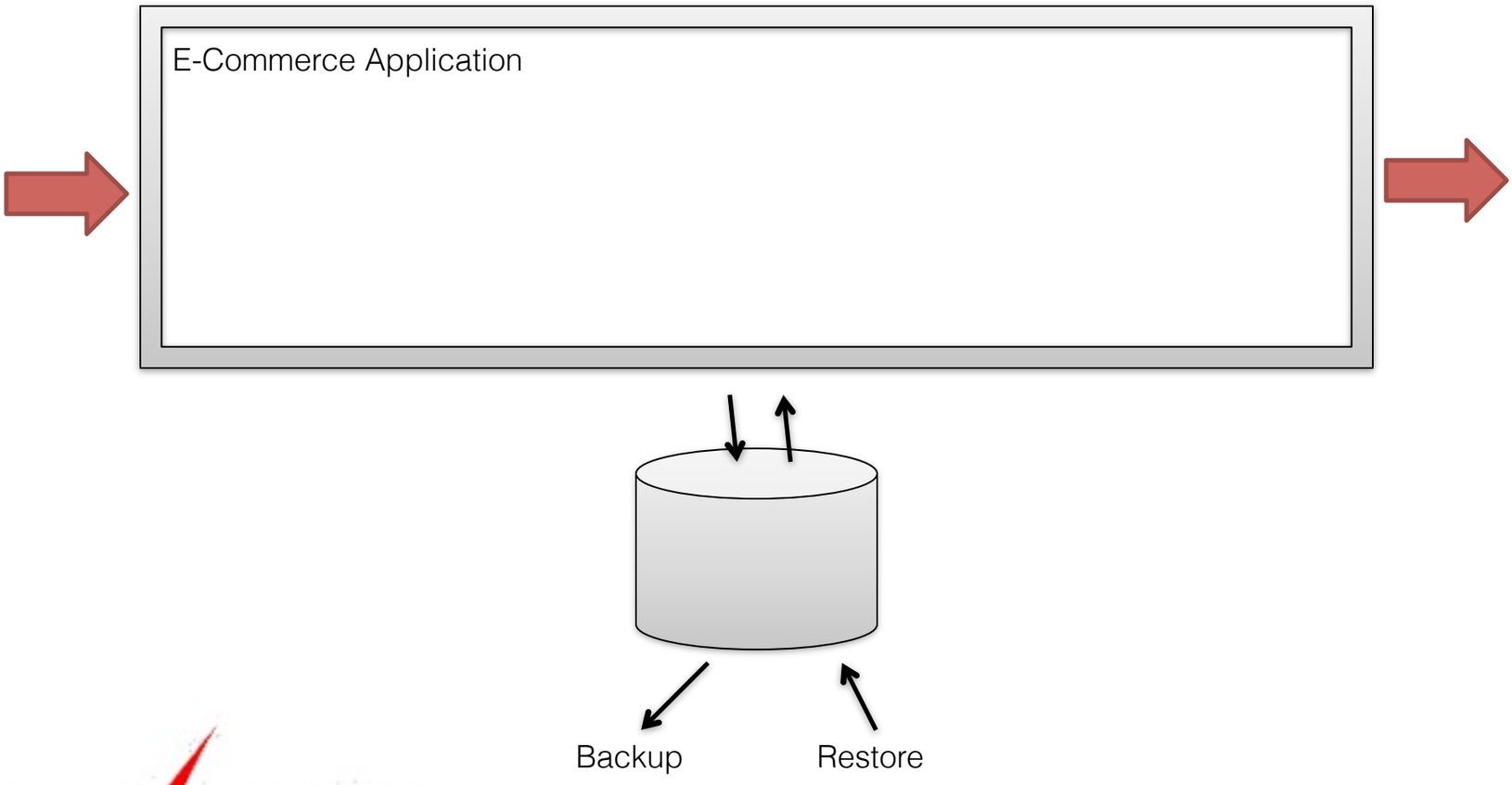

Module: Check Inventory
class InventoryCheck{
public static void main(String[]
args) {
int lo = 1;
int hi = 1;
System.out.println(lo);
while (hi < 50) {
Test
Test
Test
Backup
Restore
Module: Place Order
class OrderPlace{
public static void main(String[]
args) {
int lo = 1;
int hi = 1;
System.out.println(lo);
while (hi < 50) {
Test
Test
Test
Backup
Restore
Module: Confirm Order
class OrderConfirm{
public static void main(String[]
args) {
int lo = 1;
int hi = 1;
System.out.println(lo);
while (hi < 50) {
Test
Test
Test
Backup
Restore

Integration Test



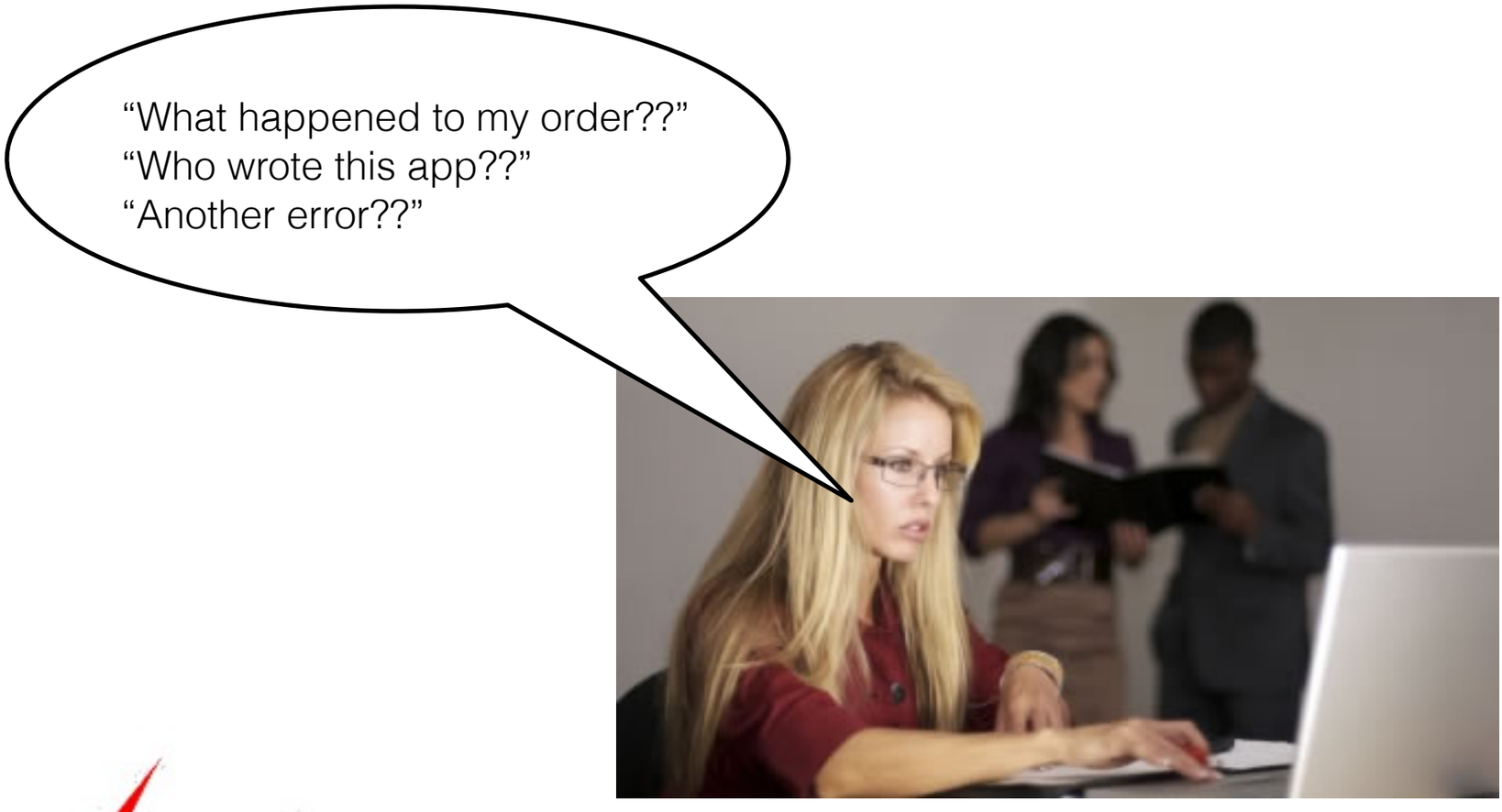
DELPHIX

System Test



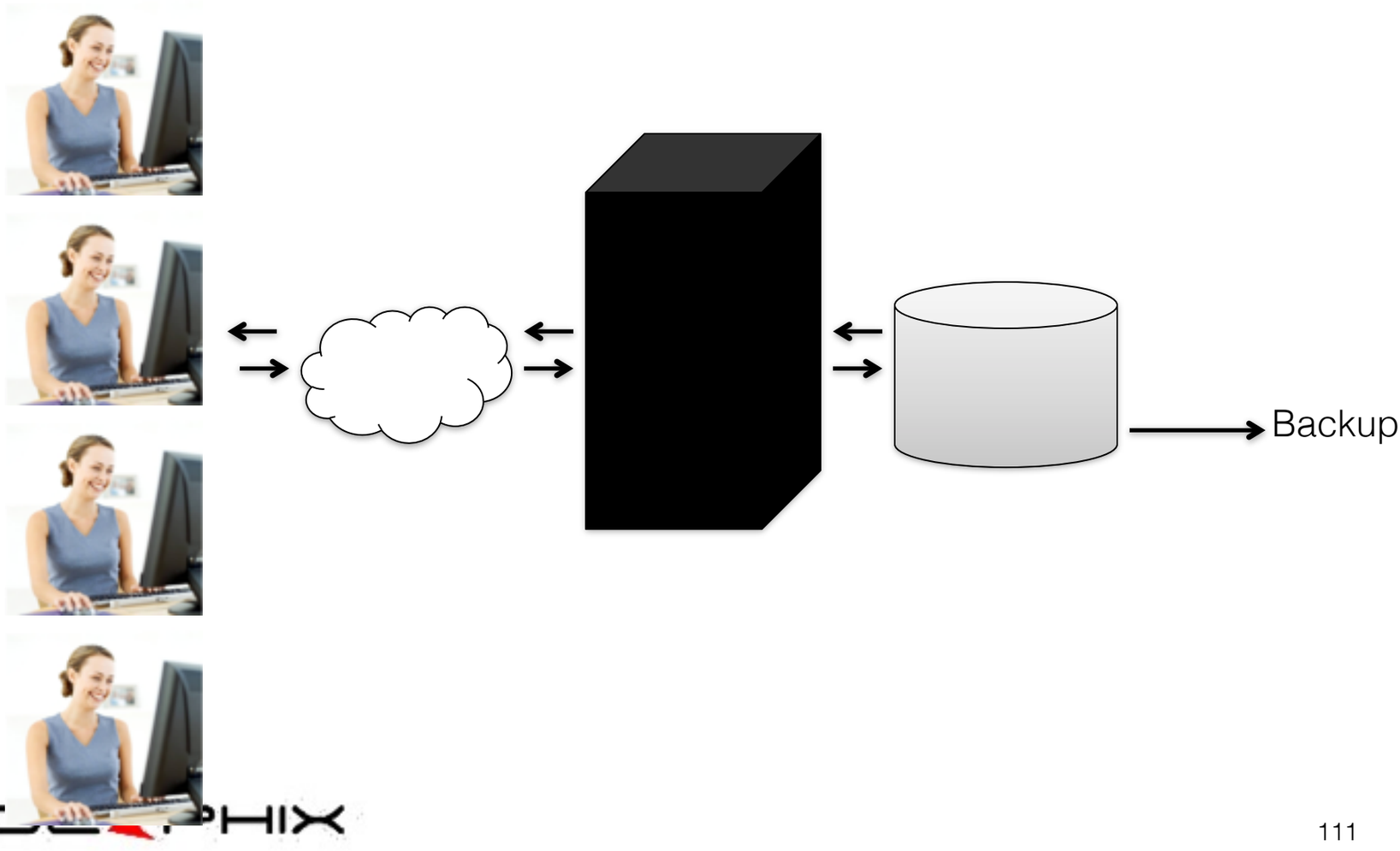
DELPHIX

User Acceptance Test



DELPHIX

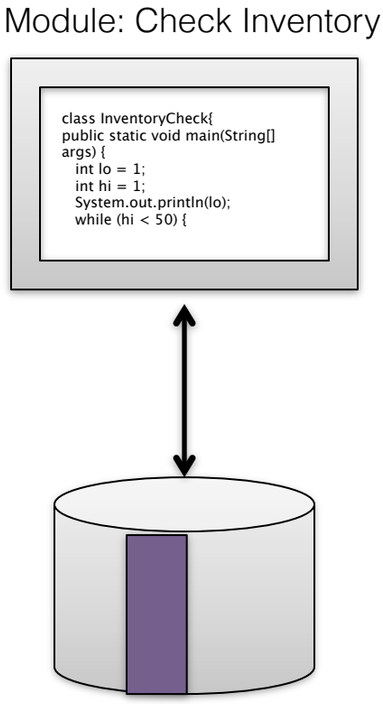
Production



Maintenance

Error!

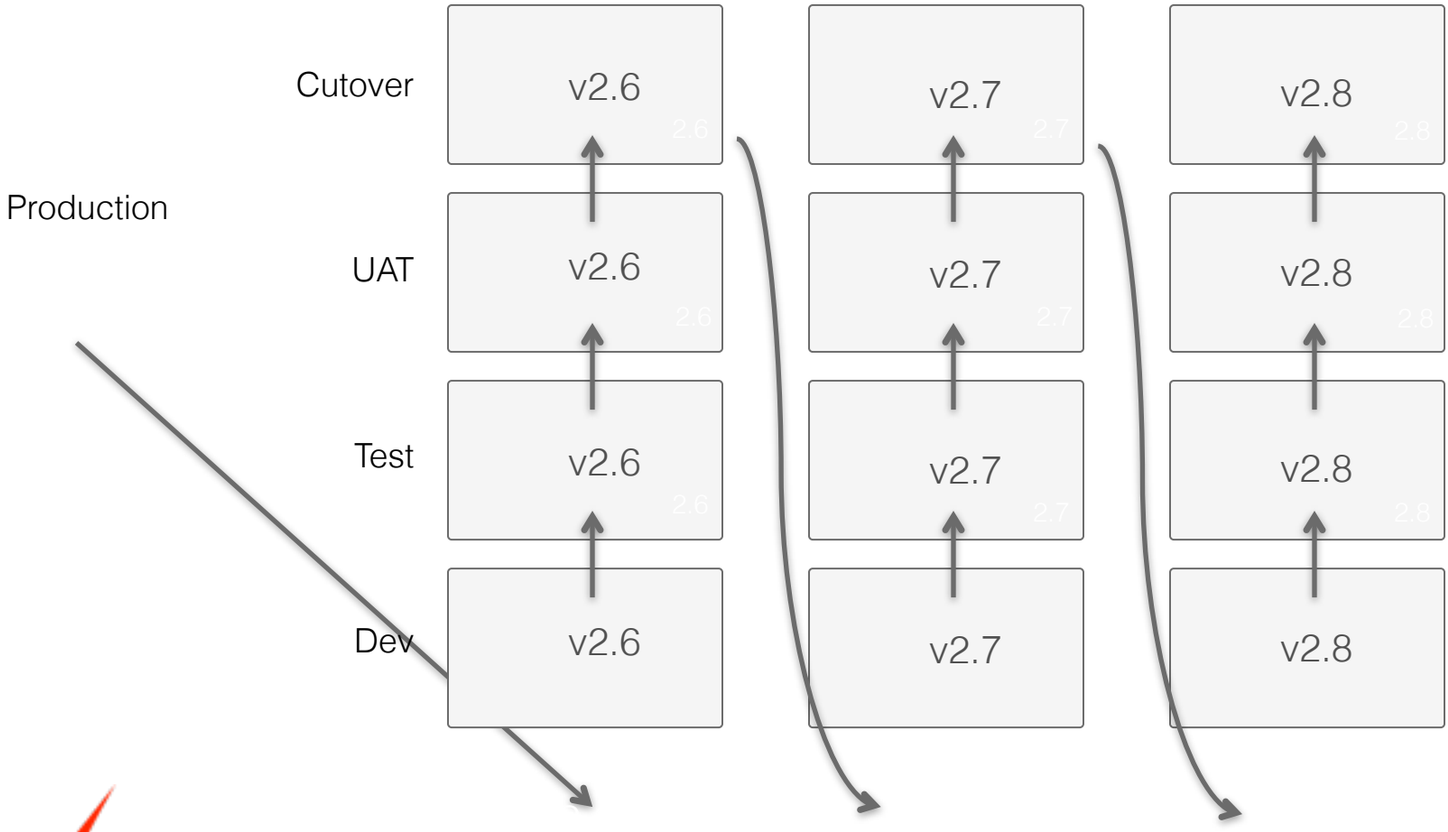
Bug Fix



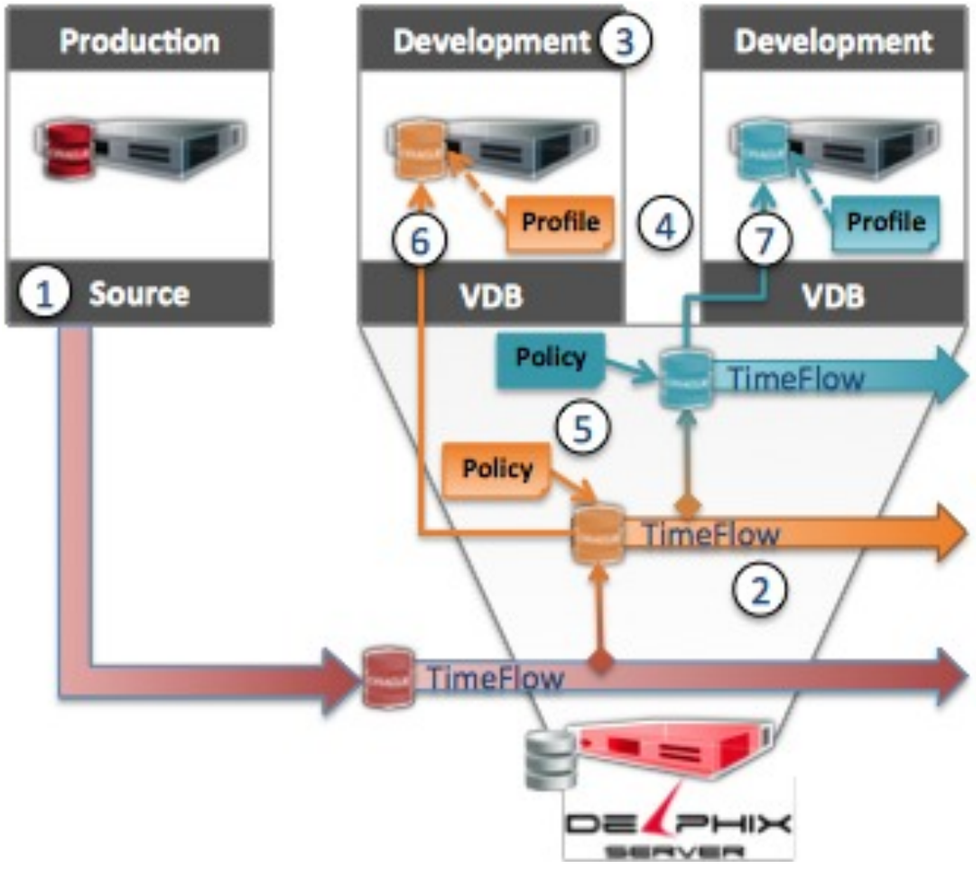
Application Version

Production

Application Version



Provisioning



DELPHIX®

The traditional approach to creating copies of database involves the restoration of back-ups or a dump to physical environments. This approach requires the full amount of storage for each copy. In addition, the copy process can tax already busy networks and overwork operations staff.

Delphix allows for the rapid provisioning and refreshing of virtual database (VDBs) without all the copying of redundant data and additional storage it requires.

The following steps show the simplicity of the Delphix approach to creating database copies:

Select the database on which the VDB will be based. For refreshes this step is unnecessary, as the source will be the same as the original VDB.

Choose the point-in-time from the TimeFlow of that database.

Select the Target Environment on which the VDB will be provisioned and associate any Pre/Post Provisioning Scripts desired.

Add the VDB Characteristics.

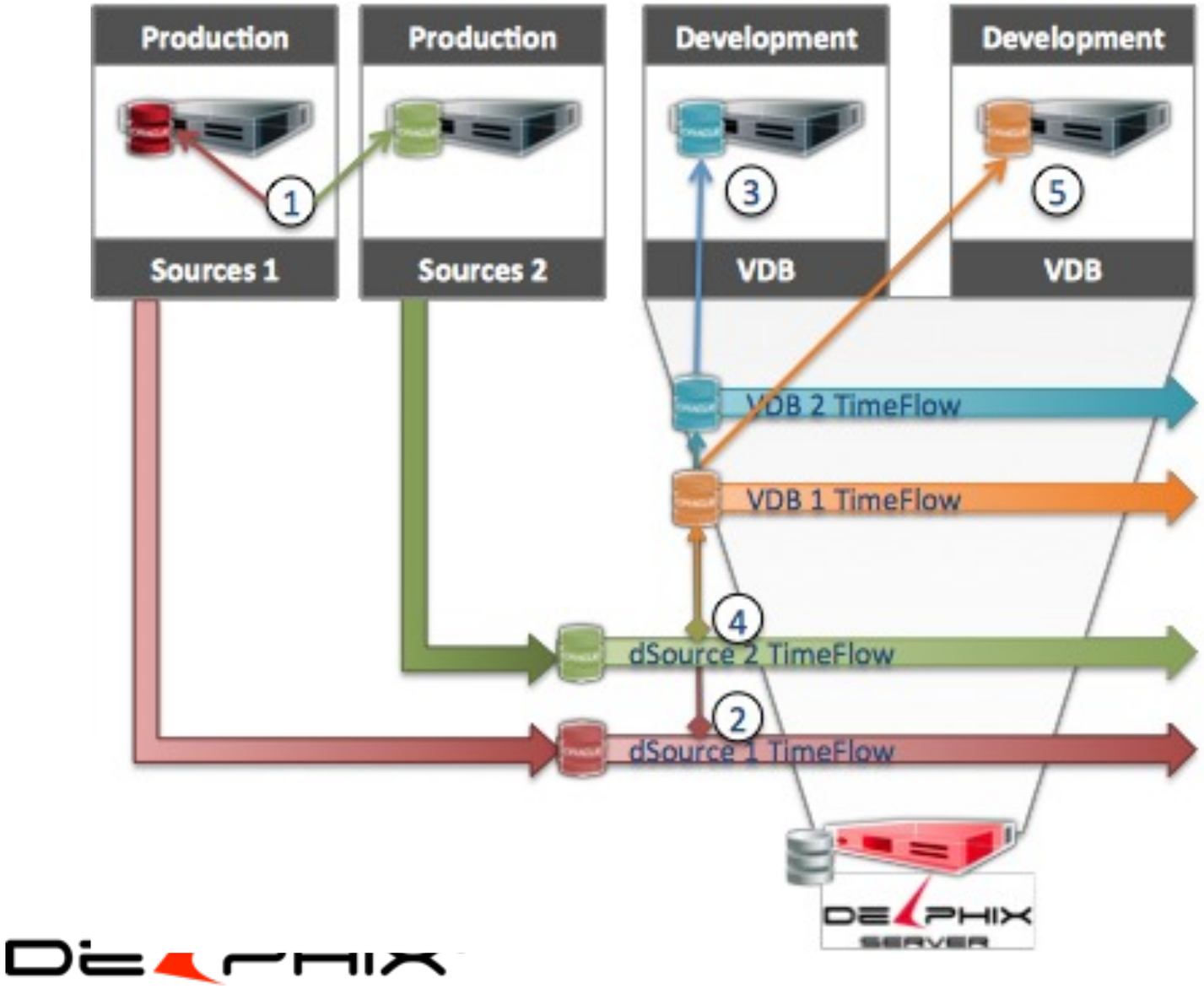
Assign the VDB to a Group and apply the SnapShot policy.

Review the input data and start provisioning.

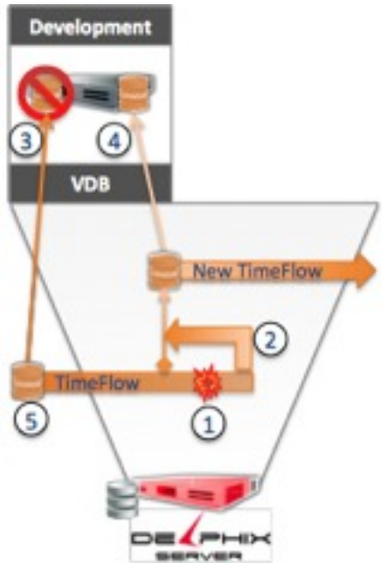
In addition, VDBs can be created from VDB is a similar fashion. Just use the VDB TimeFlow for the source of the new VDB.

Delphix fully orchestrates the provisioning/refresh process Each VDB is fully functional database, and is completely isolated from every other environment. All changes made to a VDB are only visible to that database.

The entire refresh process can be automated using the Delphix policy engine.



Federated environments have time-based dependencies between multiple databases. It is important to create test/dev environments that maintain this same data integrity. The simplest way to ensure this is to create the test environment with each database provisioned from the exact same point-in-time. You can achieve this by following these steps:
Make sure an active TimeFlow exists for each database for which there will be a copy in the target environment.
Select the Point-in-Time for the first Virtual Database (VDB).
Provision to the designated Target Server.
Select the same Point-in-Time for the second VDB.
Provision to the designated Target Server.
With Delphix you do not need to know in advance when a federated environment is needed, because it is possible to re-create federated environments from any point in time. This is especially helpful in forensic analysis and production support, when issues that occurred in the past must be recreated.



DELPHIX®

Delphix allows for the rollback of a Virtual Database (VDB) to a specific point in time or transaction. It is a built-in back-up with near instant restore, without having to deal with other back-up software, tape/VTLs, or back-up administrators. It is just as simple and intuitive as the provisioning itself.

At some point in time, there is an issue with the VDB that requires a recovery from a previous point-in-time.

Select the previous point-in-time from the TimeFlow and execute the rollback.

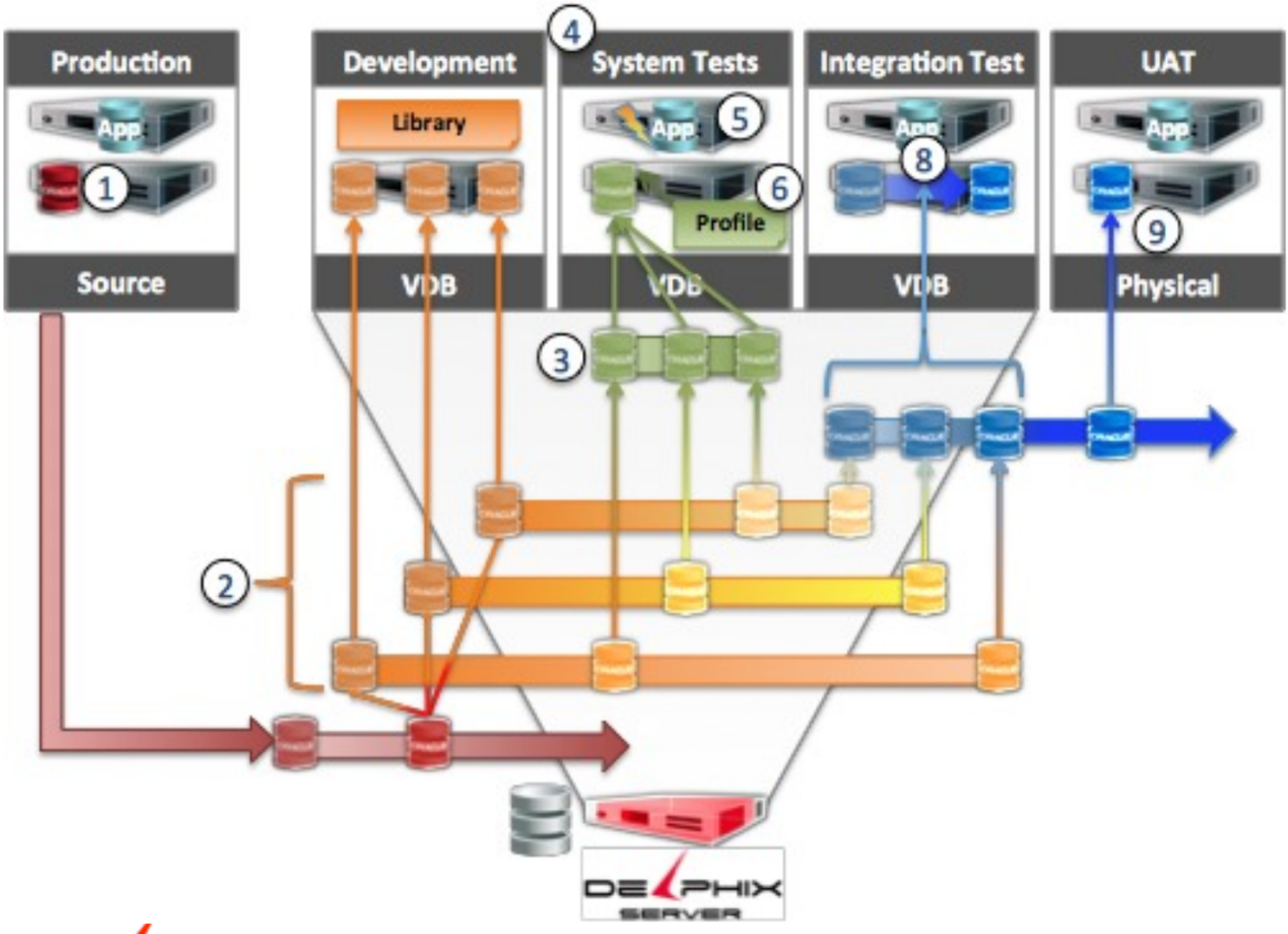
The original database will be deleted from the host.

The restored database will be provisioned to replace the original.

The TimeFlow of the original is retained in Delphix.

Because this process is performed without deleting the original TimeFlow, you can test rollbacks to multiple locations to assure that the correct point-in-time is used. You can also go back to a point before the issue and compare against the new VDB to confirm the update was successful.

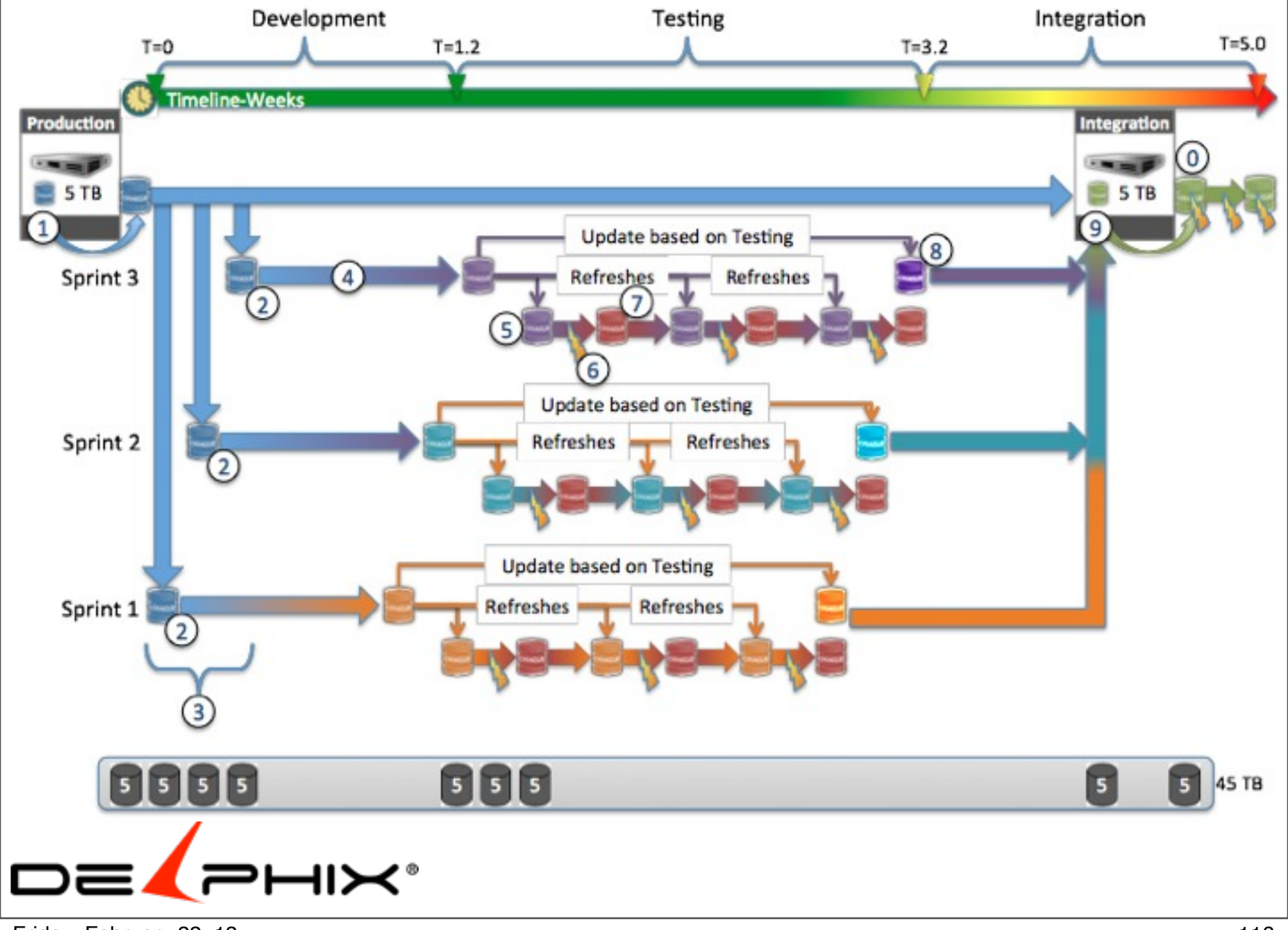
Dev



DELPHIX®

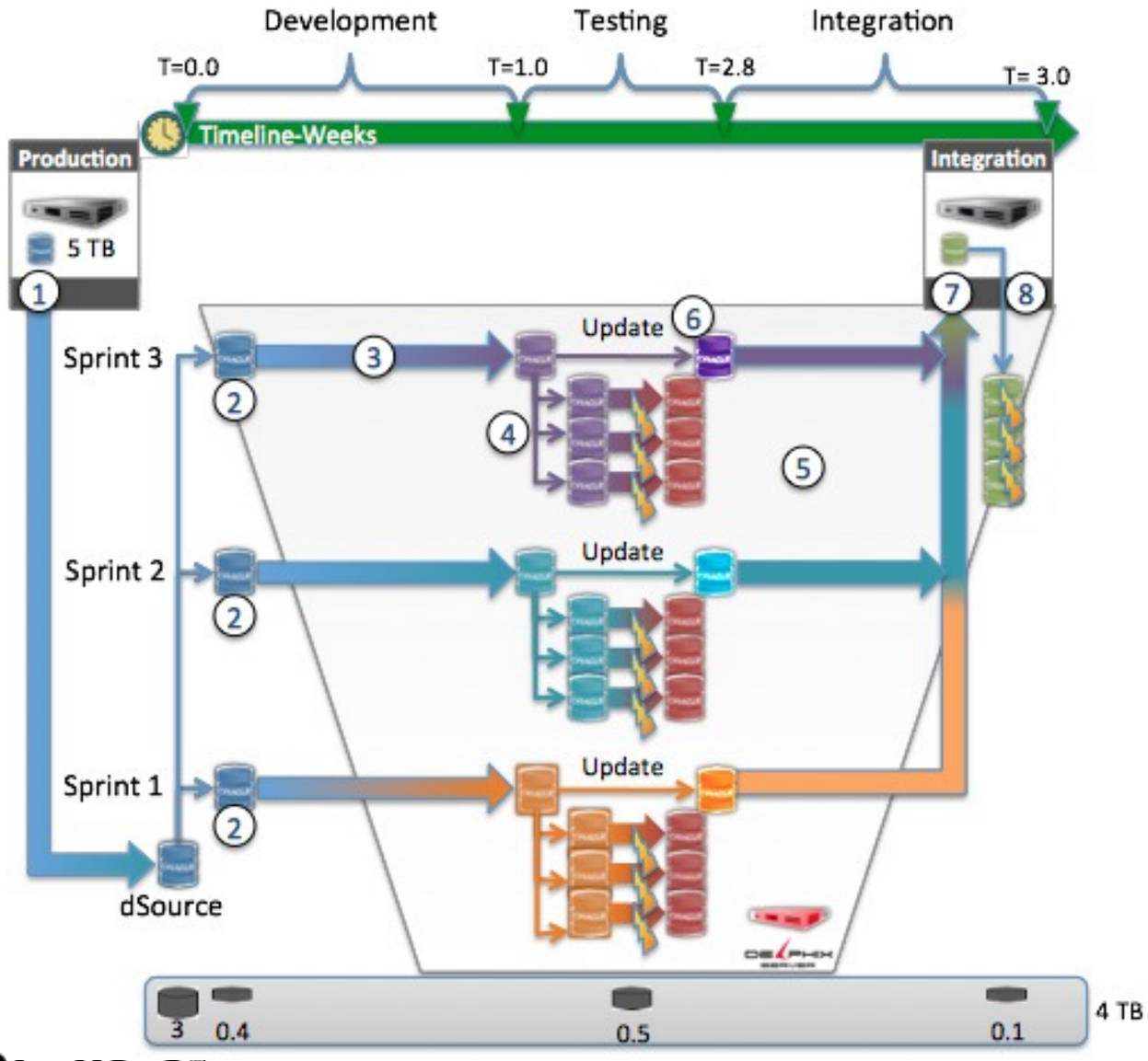
elphix can dramatically reduce the time and effort of getting the right data to the right environment at the right time. This reduces project timelines and ultimately improves product quality.
The development cycle depicted typically requires 10 or more refreshes and 1-2 days each, multiple testing environments, or a prolonged testing cycle to complete all the required testing, introduces risks in integration because of the difficulties in recovering from issues, and 10x the amount of storage of the production environment.
Here is how Delphix can be integrated into the cycle.
Synchronize the source (production) database into Delphix to create a dSource.
Create a library of Development Virtual Databases (VDBs) for each development team.
Create a complete testing environment that includes the application, middleware technologies, web servers, etc.
Create a profile for the data that will connect to the service bus of the complete application environment.
Promote VDBs from the Development Library into the testing library to test new functionality and compatibility.
Integrate the Development VDBs and test integration and functionality.
Complete the integration of the individual development project into the completed database.
Use the Delphix V2P function to create a physical copy of the completed database and perform user acceptance testing (UAT).

Agile 1

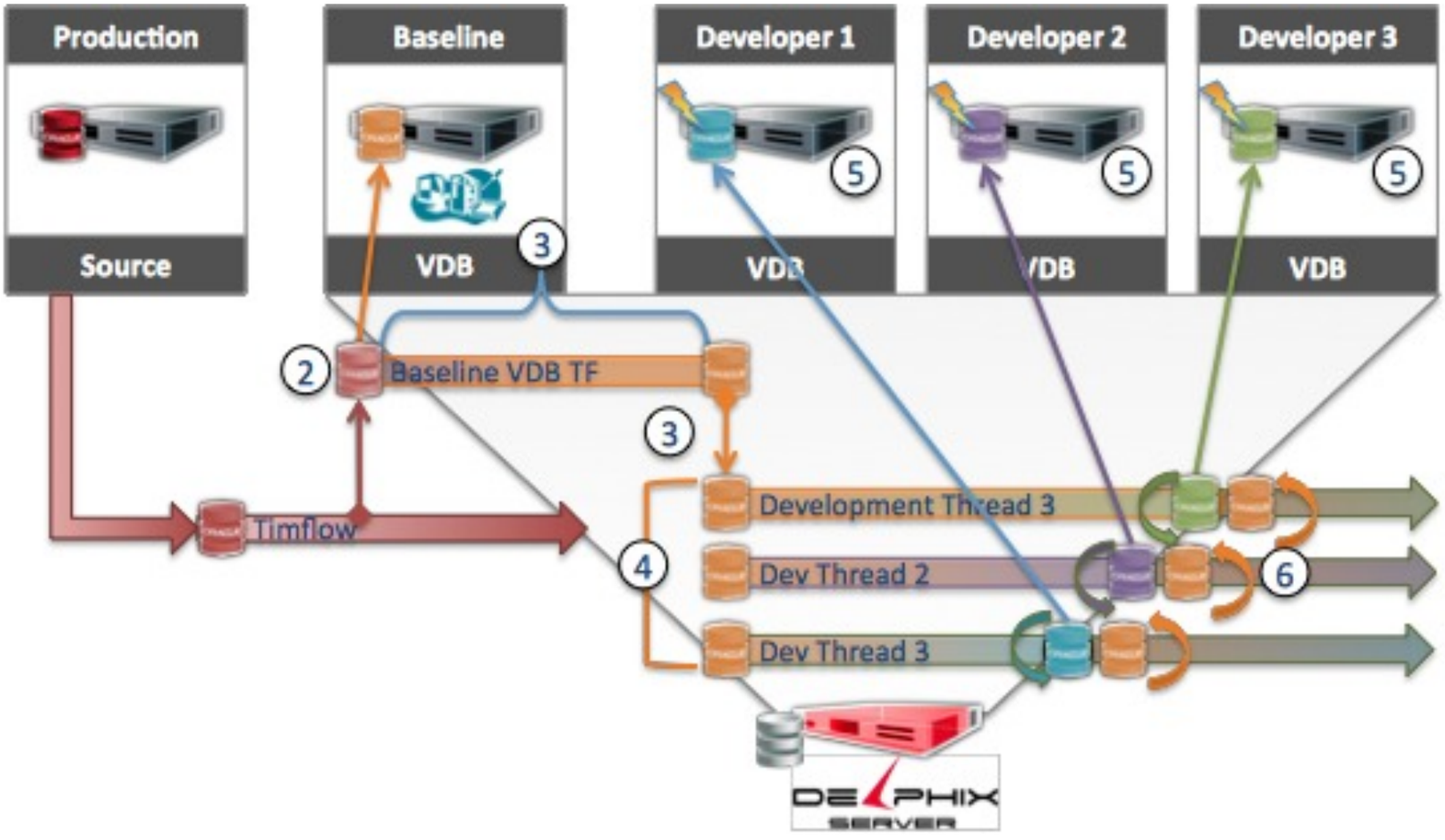


Many enterprises are turning to agile development methods in order to keep pace with the barrage of changes being requested by the business. Whether the specific method is Agile, DevOps, Scrum or some other approach, the key to success is speed and adaptability.
The main premise of these methods is to look at change as a continuous series of incremental changes. The “Sprints,” as they are often called, are designed to make smaller adjustments over shorter periods of time, typically just a few weeks. This in contrast with traditional Waterfall approaches that aim for making more dramatic changes over longer periods, often with projects running multiple years
Shorter timelines put greater pressure on operational teams that are responsible for getting the fresh data to develop against. The Delphix Agile Data Platform is designed to eliminate the complexity, wait time, and supporting infrastructure and can dramatically improve productivity.
Consider the example depicted. There are three independent sprints being conducted concurrently.
A copy of production is created as a baseline for each Sprint.
Each Sprint requires a database for development.
Creation of the development environment is a serial process because it uses the same source. This can take hours or days, depending on the database size and available infrastructure.
Development is conducted against those databases.
Testing database are created from the development copy.
Tests are conducted.
Testing environments are refreshed for another round of tests.
Development databases are updated based on test results.
The new developments are combined into an integration environment.
Integration tests are conducted.
This approach takes 5 weeks and requires 45 TB of non-production storage. The project has considerable pressure on the timeline because of the effort required to maintain fresh data for both development and testing. This often results in compromises to both development and Integration testing, and subsequent project delays and poorer quality products.
Now consider the same scenario using Dephix:
The production database is synchronized into Delphix.
Development Virtual Databases (VDBs) are created concurrently and in minutes.
Development is conducted against the VDBs as any other database.
Testing VDBs are created off the development VDBs in minutes instead of hours or days.
Tests are conducted in parallel because storage is no longer a constraint.
Development VDBs are updated based on test results.
Development VDBs are combined into an integration environment.
VDBs are created from the integrated VDB for integration testing.
The approach with Delphix only requires 3 weeks and 4 TB of storage. The savings mean that more thorough development and integration testing is conducted to assure a higher quality product. The time savings also means that the business can adapt new functionality quicker, and adapt more effectively to changes.

Agile 2


Development
Testing
Integration
T=0.0
T=1.0
T=2.8
T= 3.0
Timeline-Weeks
Production
5 TB
Integration
1
2
3
4
5
6
7
8
Update
Sprint 3
Sprint 2
Sprint 1
dSource
3
0.4
0.5
0.1
4 TB

Parallel Dev



DELPHIX®

A large part of the effort in **development** is preparing the environments with fresh data while maintaining security and confidentiality requirements. Delphix can eliminate a great deal of this effort through the use of a Baseline Virtual Database (VDB). A Baseline VDB is a database that has been prepared and secured per company requirements. It is then used as a source for the working VDBs of each development team.

The following is an example of how to create and use Baseline VDBs.

Synchronize the source database with Delphix to create a dSource.

Provision a VDB from the dSource.

Prepare the VDB to create a Baseline by obfuscating sensitive data, updating credentials for development, and other configuration changes as required.

Use the Baseline as the source for multiple VDBs, each with the desired configuration.

Each development team gets a separate VDB to develop against.

Once any of these working VDBs have been modified through user action, they can be refreshed against the baseline.

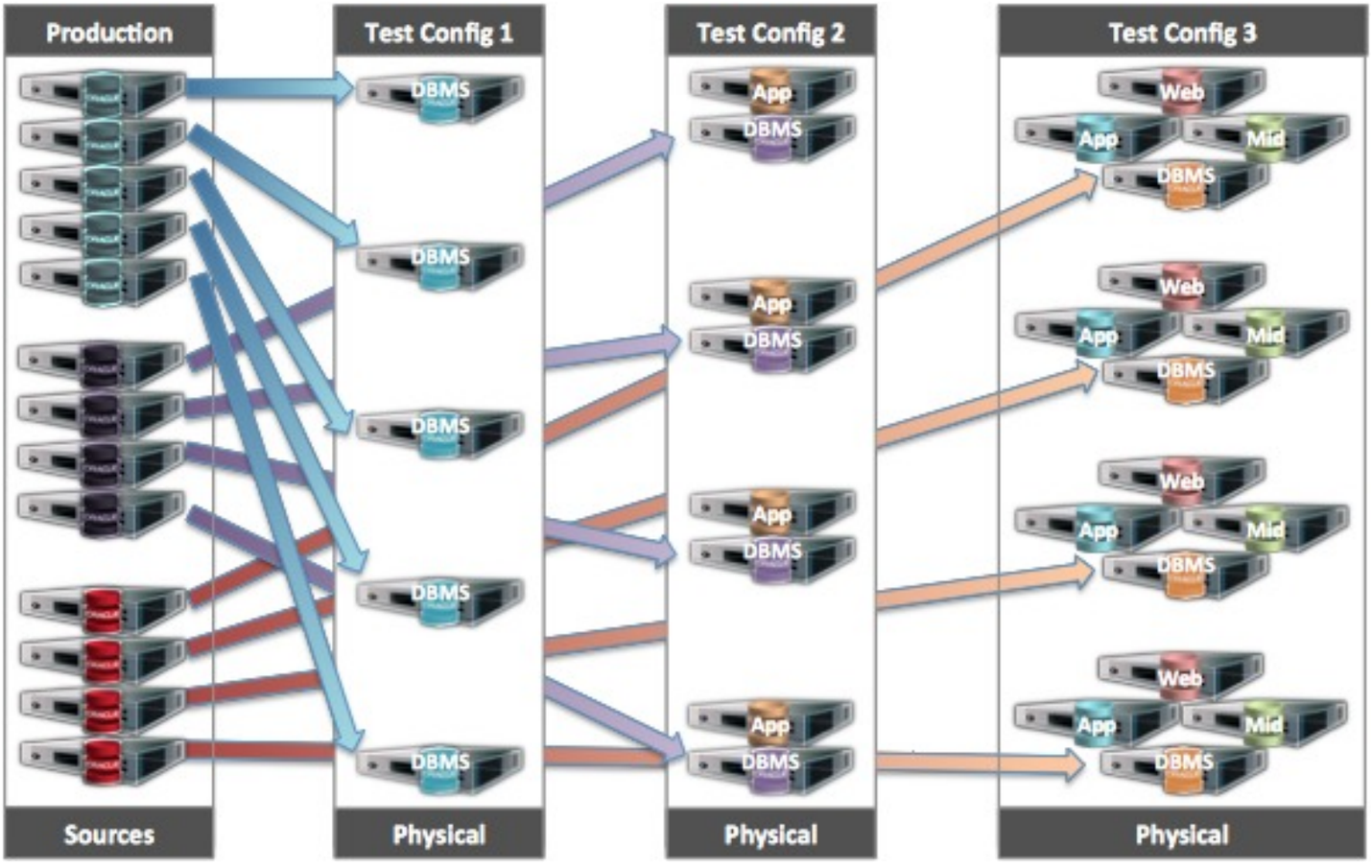
The Baseline VDB can also be used to support training and testing databases that require frequent refreshes against a common image.

Parallel **testing** efforts can be created within minutes without increasing storage requirements. This can dramatically increase the testing efficiency through more thorough testing. This leads to higher quality products.

Training environments can be refreshed against the training baseline via the Delphix policy engine. Students will automatically have a fresh database at the beginning of each exercise.

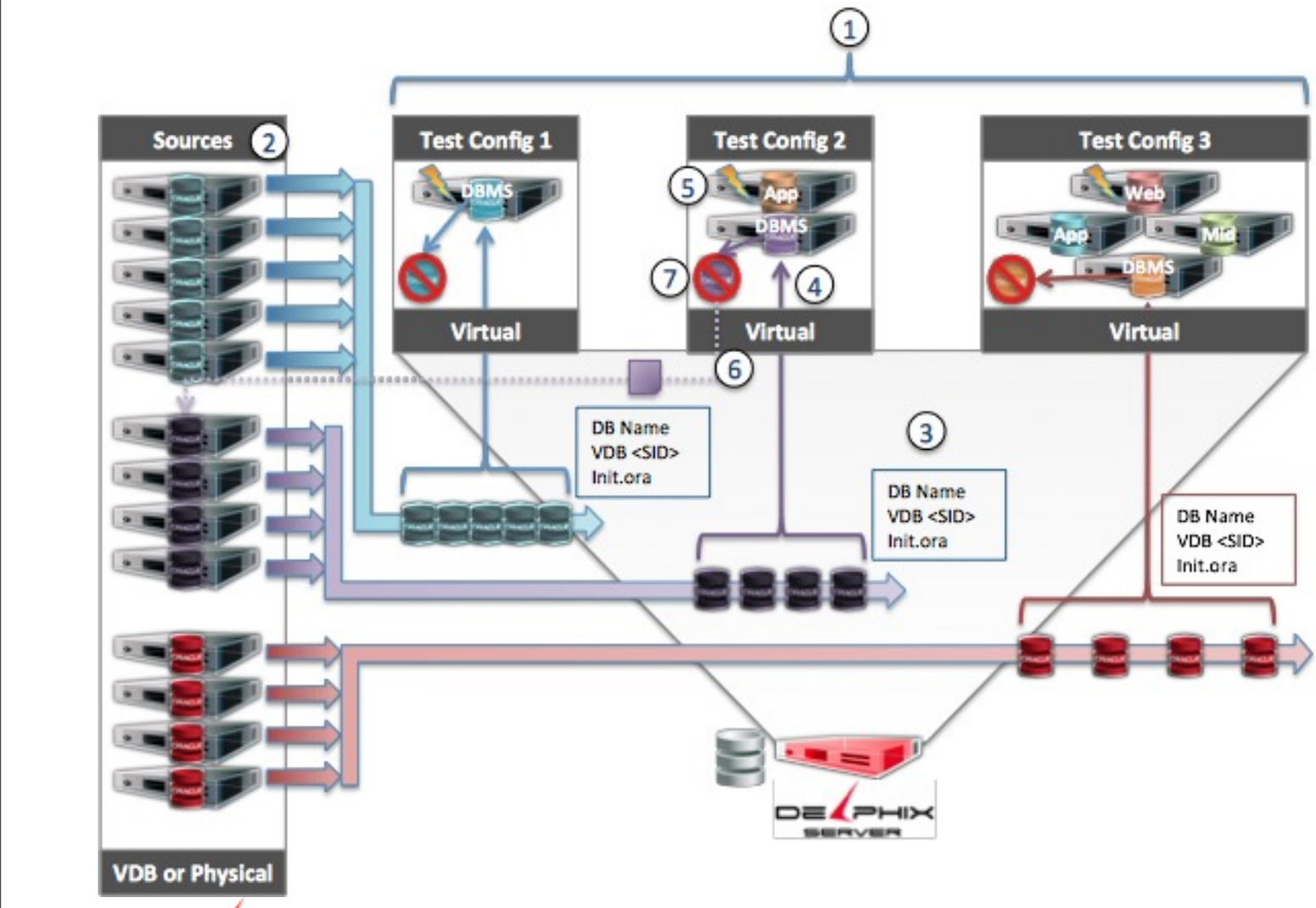
Delphix assures that all the production support environments can get a consistent database image to develop, test, or train against without the time and infrastructure costs of physical databases. .

Testing 1

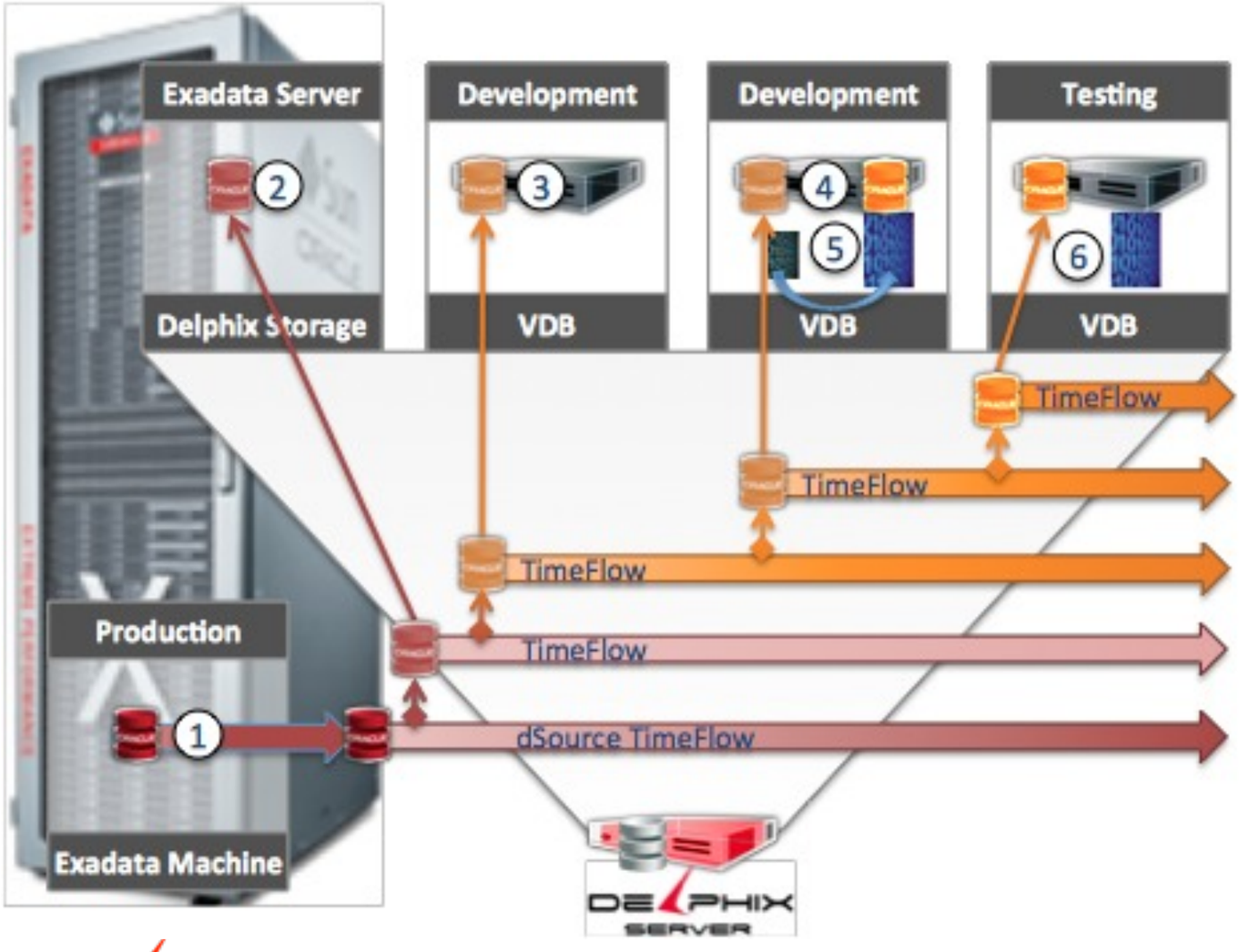


Testing is critical for improving quality and responding to incidents. Testing is conducted against copies of production or against in development process. Most organization would test more if they had the time and resources to do it.
The problem with testing with physical databases is that it can be cumbersome. Some typical problems are:
Time required to create and refresh database to testing specifications
Avoiding scheduling conflicts with other testing
Managing version control through the process
As a result, companies often dedicate testing environments. his solves some of the build and scheduling issues, but not necessarily the refresh and version control. And it creates further server and software license sprawl. This approach leads to under utilized infrastructure and software licenses as well as increasing power and cooling requirements. It is an expensive way to solve only part of the problem.
An example of this testing approach is shown on the left. There are three different types of testing environments: database only, database and application, and the entire environment including web server and middleware. These environment requires a total of 29 server to support these use cases: 13 DB server, 8 application server, 4 web and 4 middleware servers. In addition, they require a capacity equal to the e production environments.
With Delphix, you can consolidate the testing environments to only those unique configuration that need to be included in the test. An example is shown to the left using the following approach:
Create a single instance of each unique testing configurations.
Assure that all the source database are synchronized into Delphix. These can be physical database or other VDBs.
Establish a template configuration that allows the database to connect into the service bus of each of the testing configurations.
When a database is ready to be tested, create a VDB from the TimeFlow of the source, apply the template for that configuration, and connect the VDB to the testing environment.
Run the desired tests.
If there are objects that need to be updated in the source DB, they can be moved from the testing VDB to the source at the database level.
Discard the testing VDB.
This approach dramatically reduces the servers required from 29 to 6. You can also leverage your existing Delphix deployment using the same source so there may be no additional storage required.
Clients that are deploying server virtualization and/or cloud solutions can also benefit from Delphix. Virtualization and the cloud can be used to create the testing environments and Delphix is used to fill them with the data they need to test against.

Testing 2



DELPHIX®



DELPHIX®

he Exadata platforms come configured with a set amount of servers and storage. This means that you can run out of storage before you run out of server capacity. The Delphix Server is hardware agnostic, and a great way to reclaim that excess server capacity with Virtual Databases (VDBs).

Synchronize the Delphix Server to the database on the Exadata server as with any other Oracle database.

Provision a VDB back to Exadata via NFS.

Provision a VDB to other hosts, and baseline as needed.

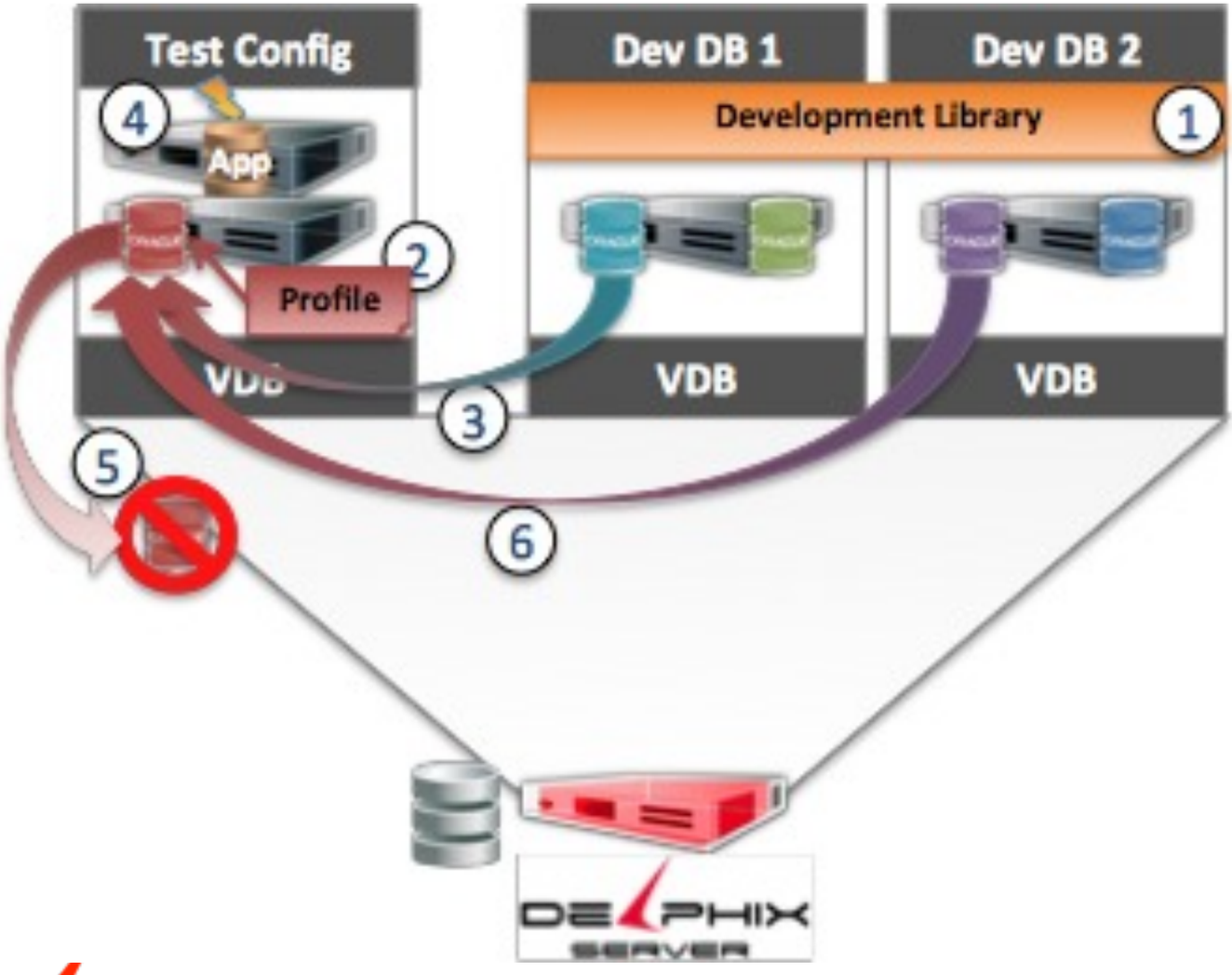
For tables that use hybrid columnar compression (HCC), provision a VDB.

The HCC compressed tables are uncompressed into new tables in the DB. All the VDB data will still be compressed in the DxFS.

Provision a new VDB from the uncompressed VDB.

VDBs can now be created for both Exadata and non-Exadata hosts.

Note that Oracle restricts licensing of some functionality, such as Smart Scan, to their hardware platforms. These functions are not available on Delphix VDBs.



DELPHIX®

Development can often be slowed down by the lack of availability of viable testing environments. There is is more to the business tha just the database, and all of the technologies involved in delivering the service must be tested together: middleware, applications, web-servers, etc. This entails a lot of infrastructure to support all those complete environments, and they are often under-utilized because of the preparation time required to test.

With Delphix, you can improve the utilization of the test environment and the ease of testing through the creation of a library of development databases. A common, complete test environment is created and used almost continuously because the preparation time of piecing it together is eliminated.

Create a “library” of Dev Virtual Databases (VDBs). These VDBs can be created from a common baseline, or a variety of sources. Each developer can have his or her own copies to develop against, without impacting other developers or projects.

Create a profile to characterize the Test DB that connects to the complete environment and save as a Delphix Template.

“Check-out” a VDB from the library by provisioning a VDB to the test DB server using the Delphix Template. This approach assures that the new VDB will connect to the environment without all the traditional preparation time.

Conduct testing against the complete environment and record the results.

Delete the test VDB.

"Check-out” another VDB in the same manner, and conduct another set of tests.

This approach simplifies and creates consistency in the entire testing approach. It becomes much easier to test early and test often.