



REAL-WORLD DATAGUARD

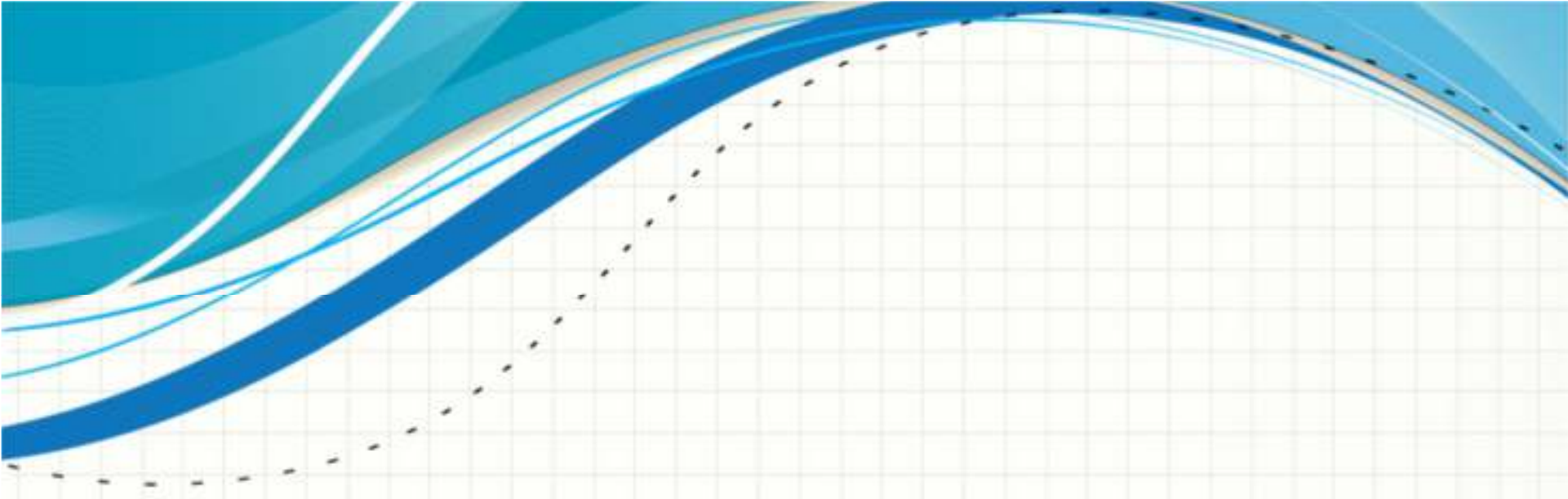
YAVOR IVANOV

SKRILL DATABASE SOLUTIONS TEAM

whoami



- 15 years experience with Oracle Database
- Oracle Database & Oracle RAC fanatic
- Oracle Certified Master
- Part of Skrill Database Solutions Team
- <http://blog.yavor.info>



Experience is the best teacher.
Enrolling in the class is the biggest problem

Disclaimer

- This presentation is strongly based on the experience of Skrill Database Solutions Team
- It is accurate in Oracle 11.2 – may be different for other versions
- It is accurate in our environment - may be different for other envs

Why we use DataGuard

- Skrill is a trusted global payment provider
 - Losing data is unthinkable
 - Downtime is very unwanted
- DataGuard gives us
 - Data loss prevention
 - DR protection
 - Greatly reduced downtime
 - Report offloading

Why we use DataGuard (2)

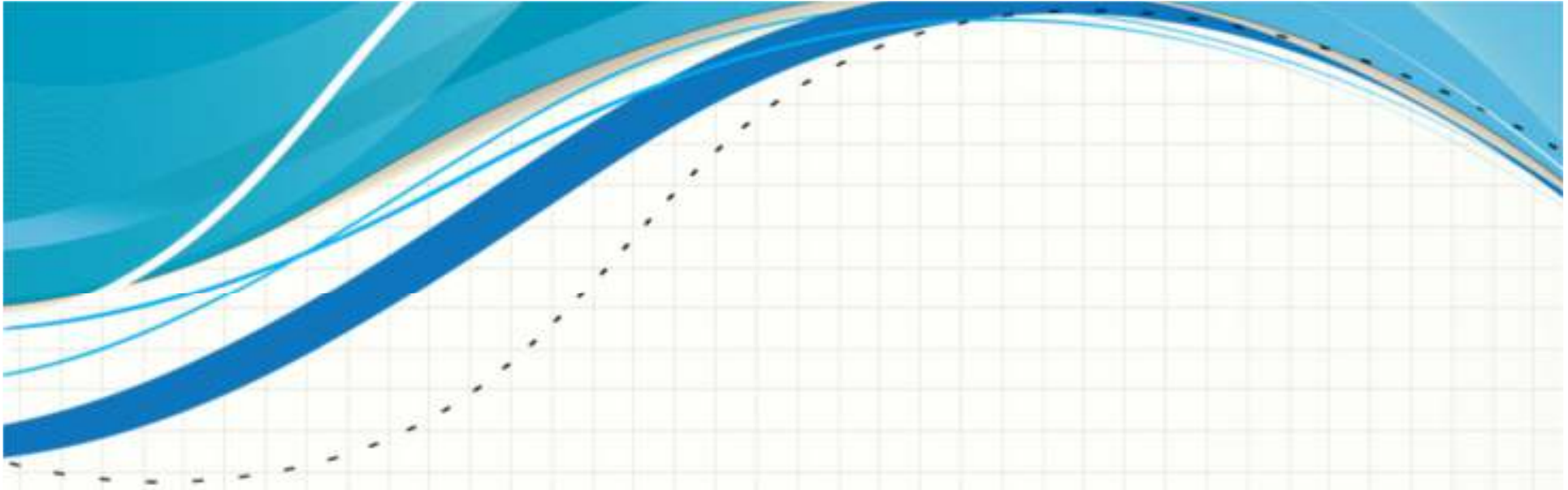
- DataGuard is
 - Easy to setup
 - Easy to support
 - Reliable
- DataGuard greatly reduces network traffic compared to storage replication
- DataGuard reduces disk IO compared to other replication tools (does not read data on primary)
- DataGuard gives us new opportunities

DataGuard drawbacks

- License cost
 - Runs only on Oracle EE
 - Requires full license also for standby DB
- All DBs have to be on the same version
 - Complex upgrades
- All or nothing – cannot replicate part of the database (except some tricks with logical standby)

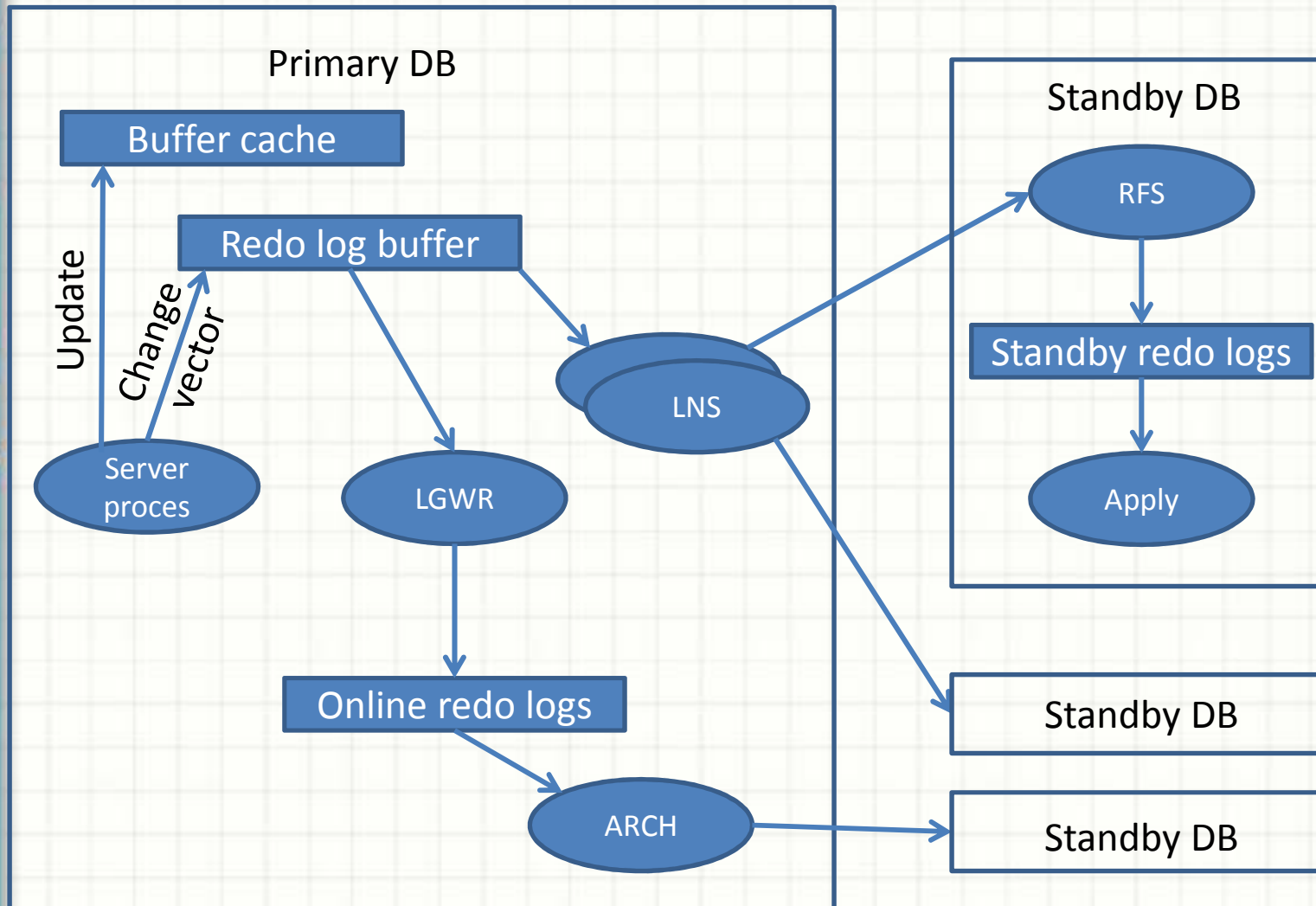
What kind for DataGuard we use in Skrill

- Physical and Logical
- Sync, Async and Lagged
- Active DataGuard
- Snapshot standby
- Cascade
 - Physical -> Physical
 - Logical -> Physical



PHYSICAL STANDBY

DataGuard concept



Physical DataGuard

- Rock solid!
- `standby_file_management=auto`
 - Creates datafiles for you, when those are created on the primary
- `db_file_name_convert` – lets you change the datafile location
 - Change diskgroup or directory
 - Change NFS to ASM

DataGuard protection modes

- Maximum Performance
 - No sync standby
 - No impact on **availability** of primary DB
 - Nearly no impact on **performance** of primary DB
- Maximum Availability
 - Sync standby *“if possible”* – `commit` completes (on primary) when data is stored on standby
 - No impact on **availability** of primary DB
 - Can impact primary **performance** in some cases
 - Zero data loss in most cases
- Maximum Protection
 - Sync standby *at all costs*
 - Affects primary **performance** and **availability**
 - Guarantees zero data loss

Sync standby and MaxAvailability

- When standby stops or becomes unreachable
 - Destination becomes ASYNC (no downtime)
- When standby gets back online
 - Standby catches up
 - Standby becomes SYNC

SYNC standby failure

(primary)

```
Wed Jun 11 08:50:05 2014
LGWR: Attempting destination
LOG_ARCHIVE_DEST_3 network reconnect
(3113)
LGWR: Destination LOG_ARCHIVE_DEST_3
network reconnect abandoned
Error 3113 for archive log file 7 to
'standby_host'
LGWR: Error 1041 disconnecting from
destination LOG_ARCHIVE_DEST_3
standby host 'standby_host'
Destination LOG_ARCHIVE_DEST_3 is
UNSYNCHRONIZED
```

(standby)

```
Wed Jun 11 08:50:05 2014
Errors in file
/.../db_lgwr_21881.trc:
ORA-04021: timeout occurred while
waiting to lock object
...
Wed Jun 11 08:50:05 2014
opiodr aborting process unknown
ospid (17685) as a result of ORA-
1092
Dumping diagnostic data in
directory=[cdmp_20140611085005],
requested by (instance=1,
osid=21881 (LGWR)),
summary=[abnormal instance
termination].
Instance terminated by LGWR, pid =
21881
```


Startup SYNC standby

(primary)

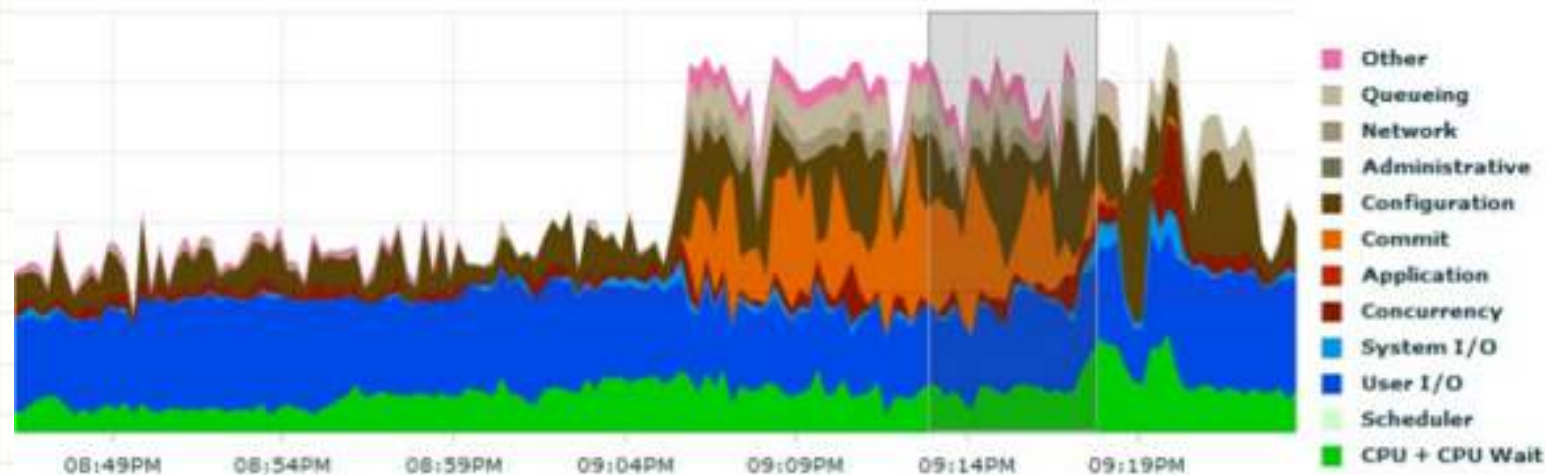
```
Wed Jun 11 08:51:58 2014
ALTER SYSTEM SET
log_archive_dest_state_3='ENABLE'
SCOPE=MEMORY SID='*';
Wed Jun 11 08:51:59 2014
LGWR: Standby redo logfile selected
to archive thread 1 sequence 495365
LGWR: Standby redo logfile selected
for thread 1 sequence 495365 for
destination LOG_ARCHIVE_DEST_3
...
Wed Jun 11 08:51:59 2014
...
Destination LOG_ARCHIVE_DEST_3 is
SYNCHRONIZED
```

(standby)

```
Wed Jun 11 08:51:43 2014
Starting ORACLE instance (normal)
...
Wed Jun 11 08:51:57 2014
Beginning Standby Crash Recovery.
Serial Media Recovery started
Managed Standby Recovery starting Real
Time Apply
...
Wed Jun 11 08:51:59 2014
Primary database is in MAXIMUM
AVAILABILITY mode
Changing standby controlfile to
RESYNCHRONIZATION level
Standby controlfile consistent with
primary
RFS[1]: Assigned to RFS process 30979
Media Recovery Log
/.../arc_495362_1_595785865.arc
...
Wed Jun 11 08:52:00 2014
...
Changing standby controlfile to MAXIMUM
AVAILABILITY level
```

Sync standby and MaxAvailability

- When network is slow, but reachable
 - Primary suffers from log file sync



- When sync standby cannot keep up writing redo (e.g. slow storage for standby logs)
 - Same as above

MaxPerformance, Redo burst and slow networks

- When big amounts of redo is generated, LNS processes cannot keep up sending the data over the slow network
- Then the primary starts ARCH processes (up to `log_archive_max_processes`)
- All those processes start burning CPU
- (and the DB has high load already)



ADG AND SYNC STANDBY

What is Active DataGuard

- Physical standby, which is in OPEN state (instead of MOUNT)
- Can service SELECT statements to offload primary
- Requires additional license

ADG connect string and failover

```
STBY_DB =  
  (DESCRIPTION =  
    (CONNECT_TIMEOUT=3) (RETRY_COUNT=3)  
    (ADDRESS_LIST =  
      (FAILOVER = ON)  
      (LOAD_BALANCE = OFF)  
      (ADDRESS = (TRANSPORT_CONNECT_TIMEOUT=2) (PROTOCOL = TCP)  
                 (HOST = adg_host)(PORT = 1521))  
      (ADDRESS = (PROTOCOL = TCP)(HOST = primary_host)(PORT = 1521))  
    )  
    (CONNECT_DATA =  
      (SERVICE_NAME = adg_db)  
    )  
  )  
)
```


How SYNC is your SYNC ADG

- **Synchronous redo transport** requires a primary database to wait for confirmation from the standby that redo has been received and written to disk before commit success is signaled to the application
- SYNC does not cover the apply
- Even for local SYNC ADG, we measured apply lag of ~100-200 ms
- Our application was not happy

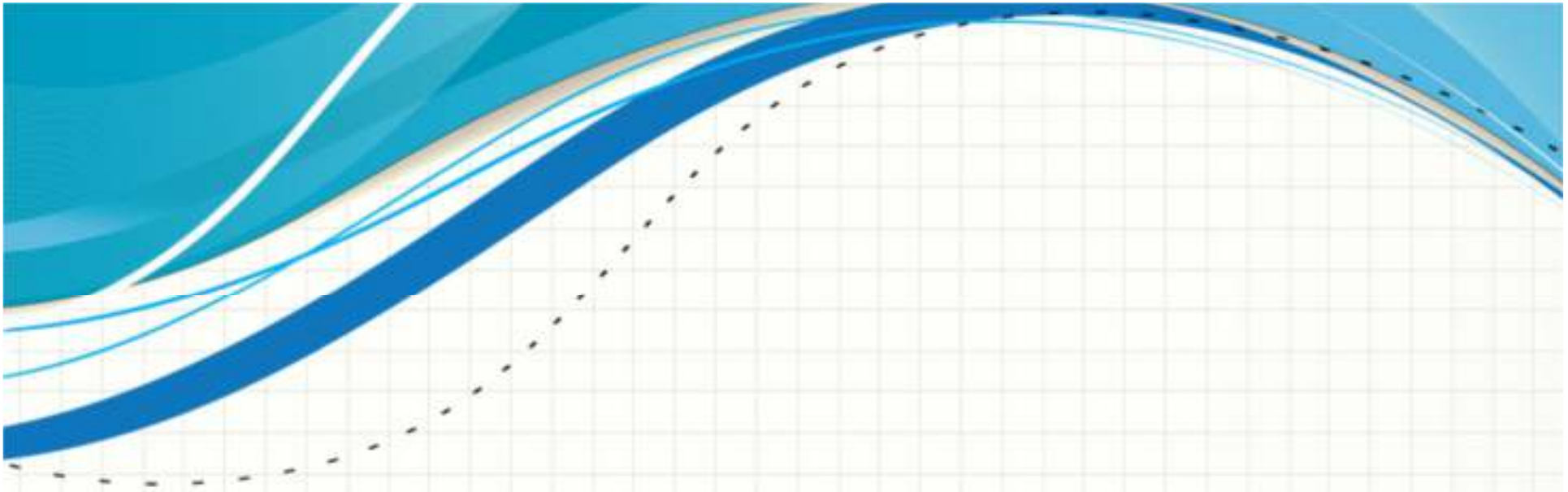
STANDBY_MAX_DATA_DELAY

- Can be set per session
- Set how much lag (in seconds) your query tolerates
- If the lag is higher than the desired value, the query returns
ORA-03172: STANDBY_MAX_DATA_DELAY
of X seconds exceeded
- The application should capture the error and either wait or query the primary

Monitor the lag

```
SQL> select name, value, unit from v$dataguard_stats;
```

NAME	VALUE	UNIT
transport lag	+00 00:00:00	day(2) to second(0) interval
apply lag	+00 00:00:00	day(2) to second(0) interval
apply finish time	+00 00:00:00.000	day(2) to second(3) interval
estimated startup time	19	second



SNAPSHOT STANDBY

Snapshot standby

- Open read-write

```
ALTER DATABASE RECOVER MANAGED STANDBY  
DATABASE CANCEL;
```

```
ALTER DATABASE CONVERT TO SNAPSHOT STANDBY;
```

```
ALTER DATABASE OPEN;
```

- Convert back to standby

```
shutdown immediate
```

```
startup mount
```

```
ALTER DATABASE CONVERT TO PHYSICAL STANDBY;
```

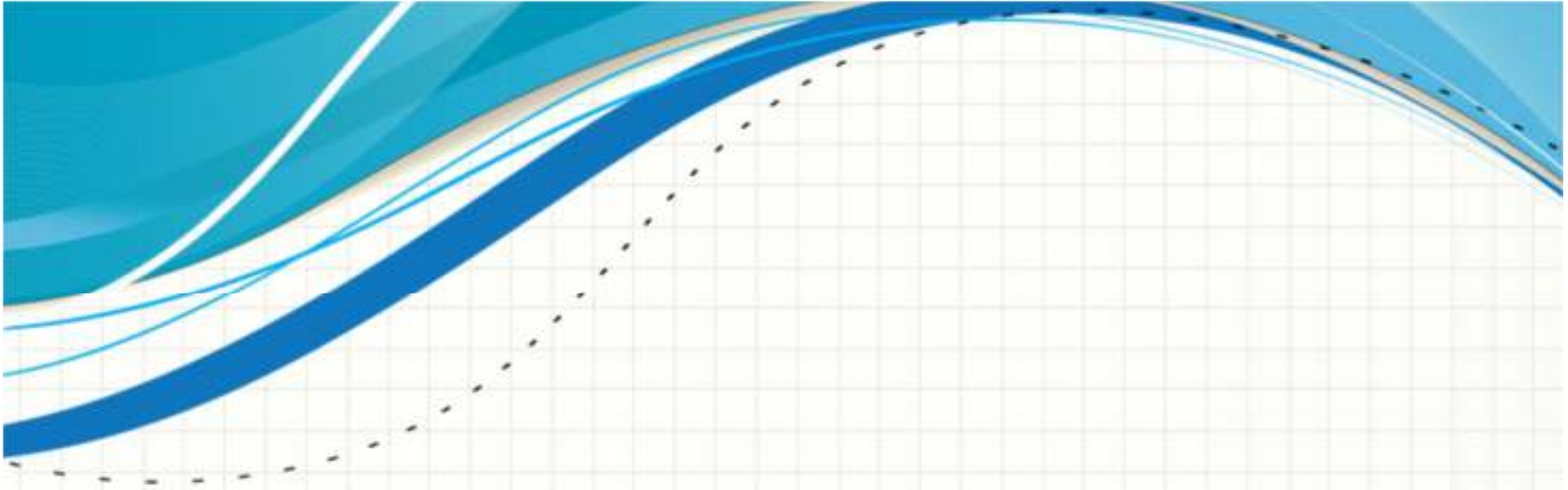
(start the apply using broker or sqlplus)

Snapshot standby

- Great for test environments
- Easy to setup
- Can be automated
- Uses flashback behind the scenes
- Can be configured to receive redo data all the time – so we have the DataGuard protection!

Snapshot standby considerations

- Do not drop tablespaces/datafiles in snapshot mode
- Do not shrink datafiles in snapshot mode
- Ensure you have enough space for
 - Flashback logs – depends on the operations you do in snapshot mode
 - Archive logs – depends on the load on primary



SWITCHOVER

Switchover

- Prepare application
 - Application should be able to run from both databases
 - Use services
- Prepare database
 - Note 1305019.1 **Data Guard Physical Standby Switchover Best Practices using the Broker**
- Use low-load time for the switchover
- Use DataGuard broker

Switchover downtime (production, real-world figures)

- Primary is available in 0:40 to 1:40 sec
- Standby is available in 2-4 minutes (*note: standby is restarted by the broker*)
- Actual downtime is 1 to 2 minutes (*with proper application setup*)

Clusterware and switchover considerations

```
$ srvctl start instance -d db_name -i db2
```

```
PRCR-1013 : Failed to start resource ora.db_name.db
```

```
PRCR-1064 : Failed to start resource ora.db_name.db on node  
server02
```

```
CRS-5017: The resource action "ora.db_name.db start" encountered  
the following error:
```

```
ORA-16002: database already open for read/write access by  
another instance
```

```
$ srvctl config database -d db_name
```

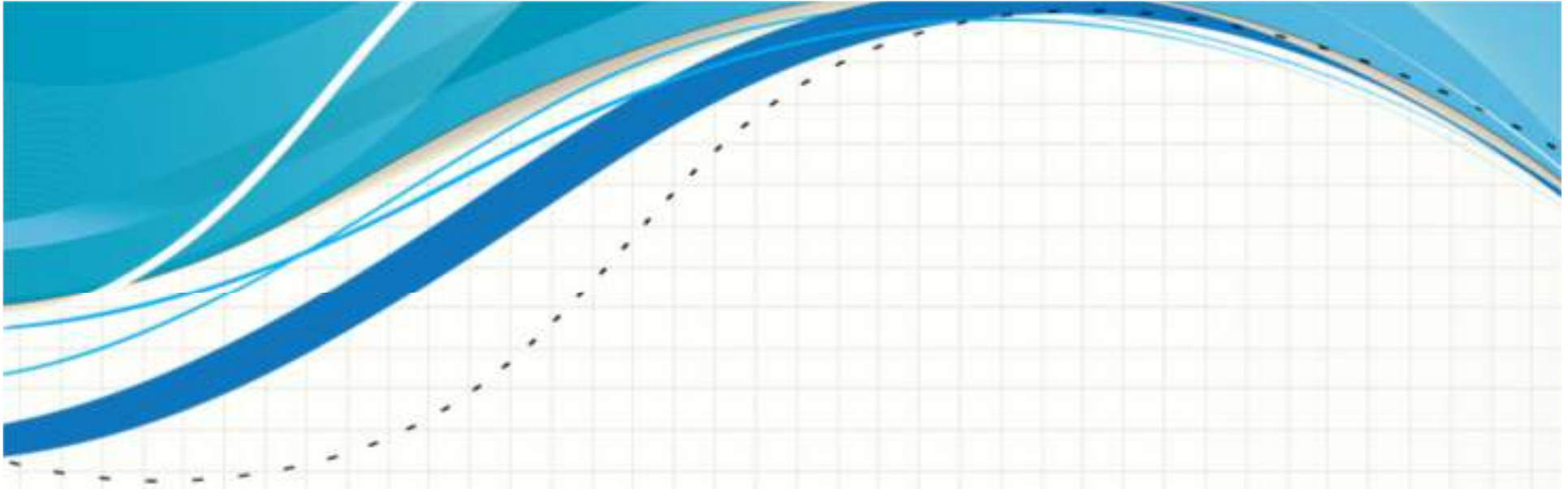
```
Database unique name: db_name
```

```
...
```

```
Start options: read only
```

```
...
```

```
$ srvctl modify database -d db_name -s open
```



FUTURE PLANS

(near) Zero downtime upgrade to 12c

- We cannot afford 30-60 minutes downtime for “regular” upgrade
- We plan to use Transient logical standby
- This is yet to be tested
- *Other option is GoldenGate*

12c: ADG enhancements for reporting

- Oracle 12c enables “temporary undo” to be written in TEMP instead of UNDO tablespace
- This allows usage of temporary tables on Active DataGuard database
- Oracle allows also using the sequences on ADG
 - Gets CACHE number of IDs from primary – ensure big CACHE for the sequences

12c: Fast sync


- Sync standby returns OK once the data is received in memory
- Faster roundtrip
- Primary will not suffer from slow disk for standby redo logs

12c: Far sync

- Allows synchronous zero-data-loss standby over long distance (*kind of*)
- Special FAR SYNC instance close to the primary, that has only control files and redo logs
 - Does not need CPU power
 - Does not need expensive storage
- Primary sends the data to FAR SYNC instance in SYNC manner. FAR SYNC sends to remote DB
- If primary fails, we still have the data
- FAR SYNC can be in other DC, close to the primary

Other 12c enhancements

- `VALIDATE DATABASE` check if configuration is ready for switchover
- Resumable switchover – when switchover fails:
 - Resolve the problem and re-issue broker switchover – broker picks up where it left off
 - Use broker to switch back to the original primary while the problem is resolved
 - Use broker to switchover to another standby database in a multi-standby configuration
- Configurable thresholds for transport lag and apply lag
 - The broker generates a warning status if transport or apply is affected in any way that creates the potential for data loss to exceed RPO
- Global Data Services – load-balancing over ADG
- Application continuity – continue transaction after failover or switchover



Before I came here, I was
confused about this subject.
Having listened to your lecture, I
am still confused, but on a higher
level

—*Enrico Fermi*



THANK YOU!

Yavor Ivanov
Skrill Database Solutions team

ORACLE®

Certified Master