



# Active Data Guard at CERN

Luca Canali – CERN

Marcin Blaszczyk – CERN



*UKOUG Conference, Birmingham, 3rd-5th December 2012*

# Outline

- CERN and Oracle
- Architecture
- Use Cases for ADG@CERN
- Our experience with ADG and lessons learned
- Conclusions



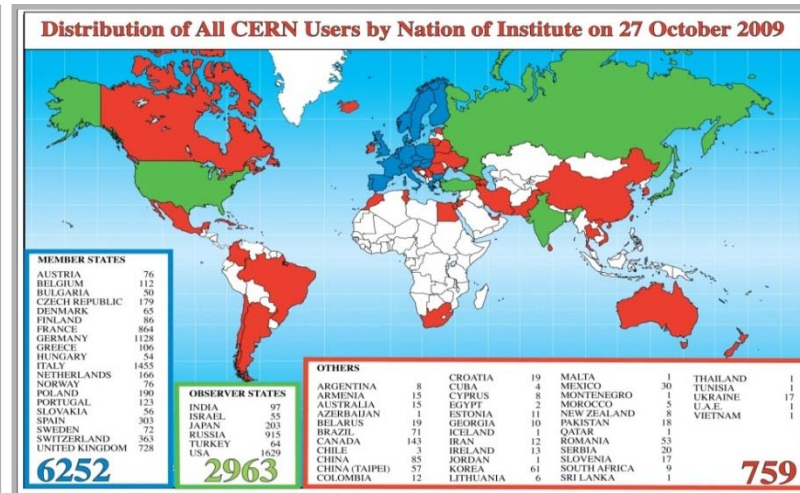
# Outline

- CERN and Oracle
- Architecture
- Use Cases for ADG@CERN
- Our experience with ADG and lessons learned
- Conclusions



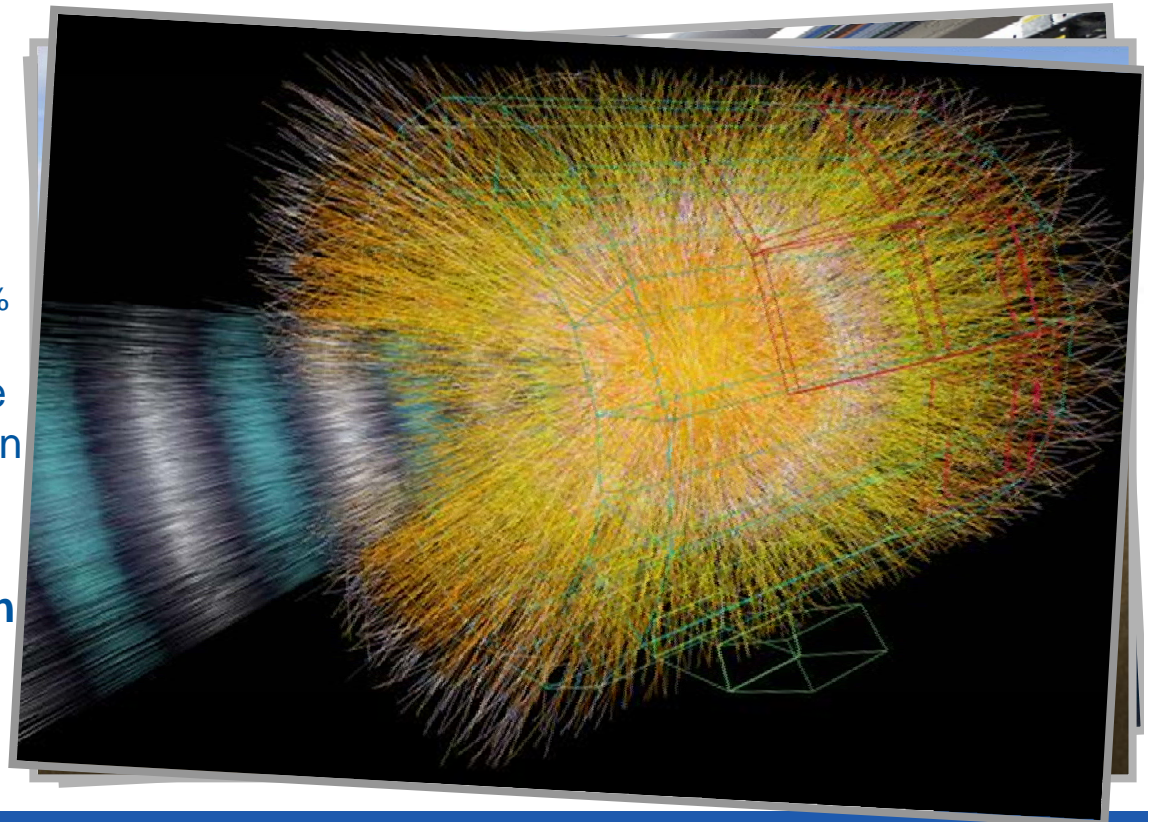
# CERN

- European Organization for Nuclear Research founded in 1954
- 20 Member States, 7 Observer States + UNESCO and UE
- 60 Non-member States collaborate with CERN
- 2400 staff members work at CERN as personnel, 10 000 more researchers from institutes world-wide



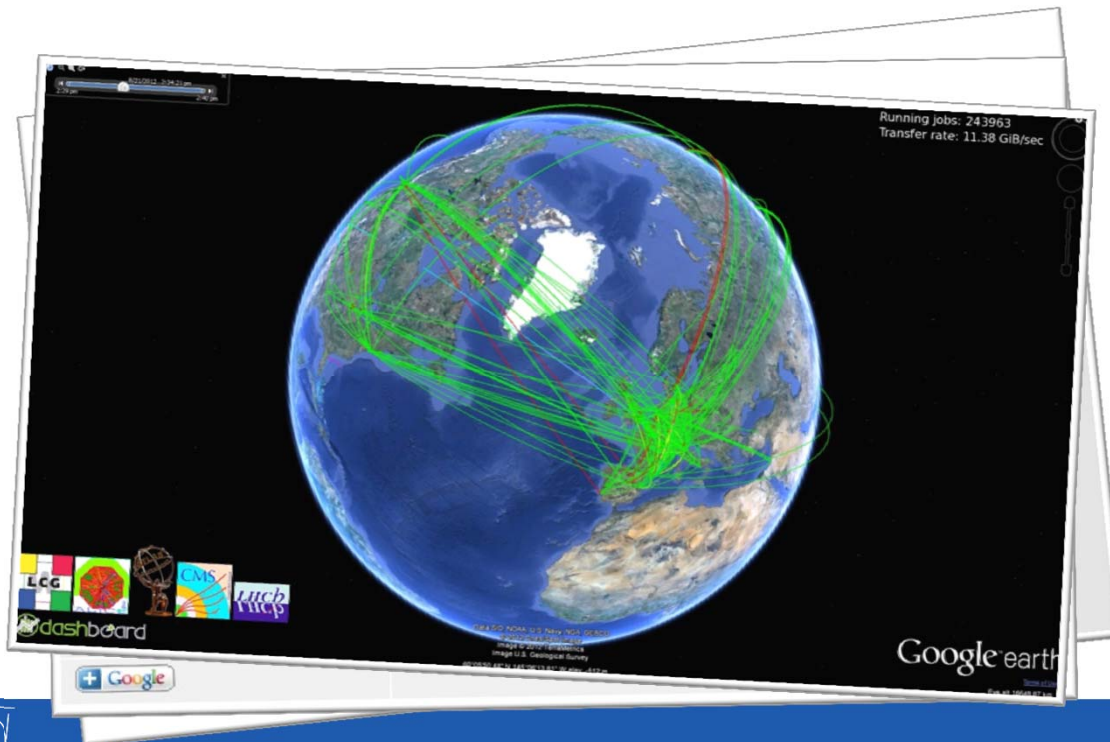
# LHC and Experiments

- Large Hadron Collider (LHC) – particle accelerator used to collide beams at very high energy
  - 27 km long circular tunnel
  - Located ~100m underground
  - Protons currently travel at 99.9999972% of the speed of light
- Collisions are analysed with usage of special detectors and software in the experiments dedicated to LHC
- **New particle discovered!**  
**consistent with the Higgs Boson**



# WLCG

- The world's **largest** computing grid



More than 20 Petabytes  
of data stored and analysed  
every year

Over 68 000 physical CPUs  
Over 305 000 logical CPUs

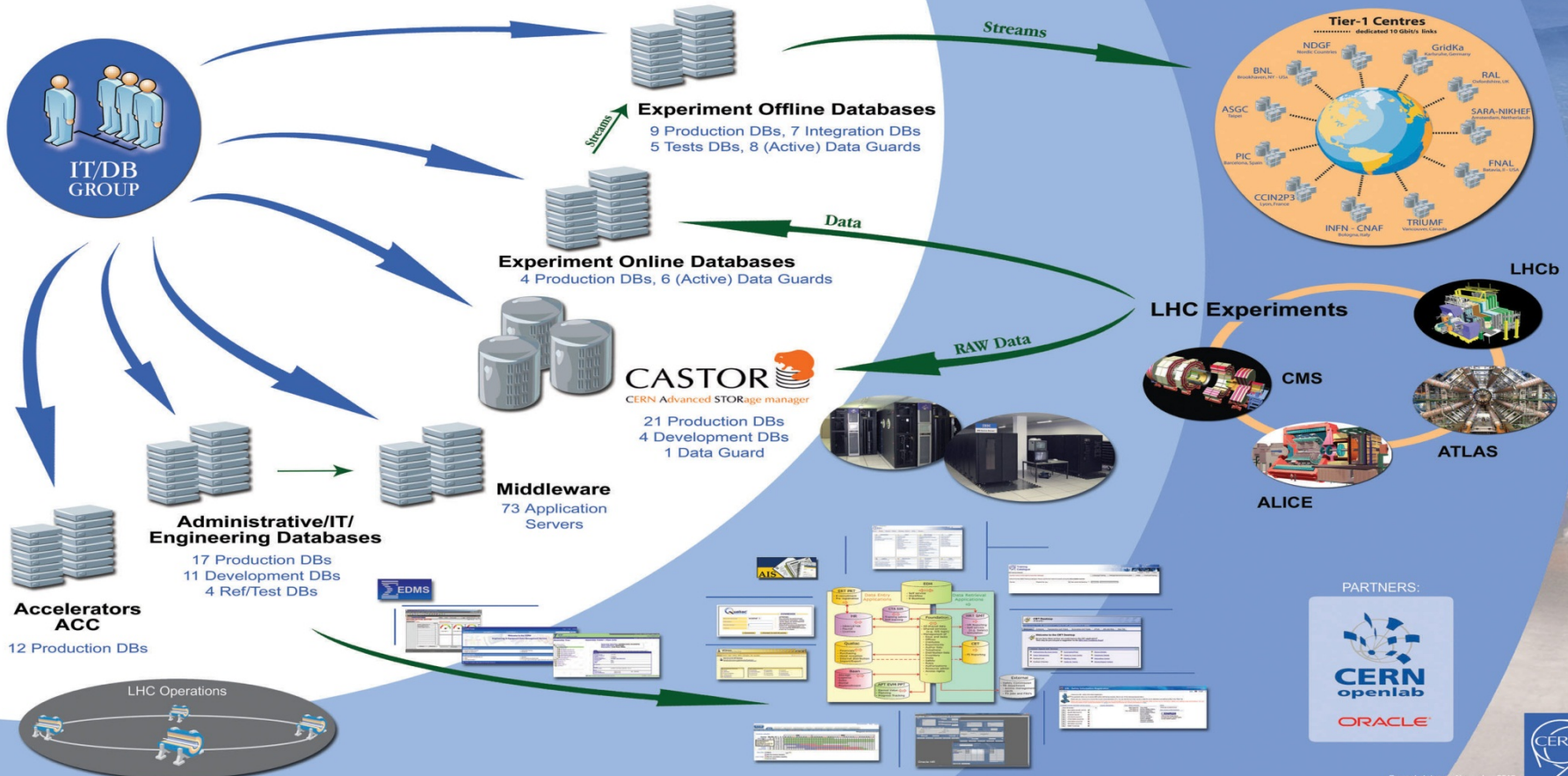
157 computer centres in 36  
countries

More than 8000 physicists with  
real-time access to LHC data

# Oracle at CERN

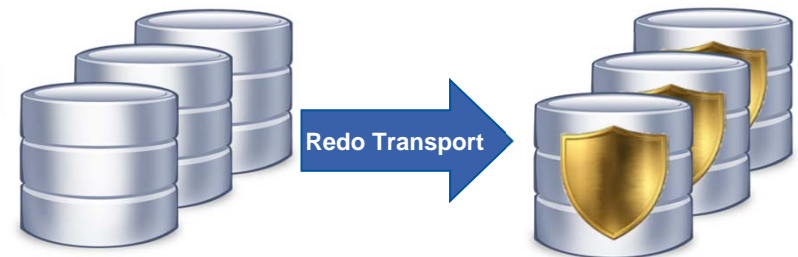
- Relational DBs play a key role in the LHC production chains
  - Accelerator **logging** and **monitoring** systems
  - **Online** acquisition, **offline**: data (re)processing, data distribution, analysis
  - Grid infrastructure and operation services
    - Monitoring, dashboards, etc.
  - **Data management** services
    - File catalogues, file transfers, etc.
  - **Metadata** and **transaction** processing for tape storage system





# CERN's Databases

- **~100** Oracle databases, most of them RAC
  - Mostly NAS storage plus some SAN with ASM
  - **~300 TB** of data files for production DBs in total
- Examples of critical production DBs:
  - LHC logging database **~140 TB**, expected growth up to **~70 TB / year**
  - 13 Production experiments' databases **~120 TB** in total
- **15 Data Guard** RAC clusters in Prod
  - Active Data Guards since upgrade to 11g

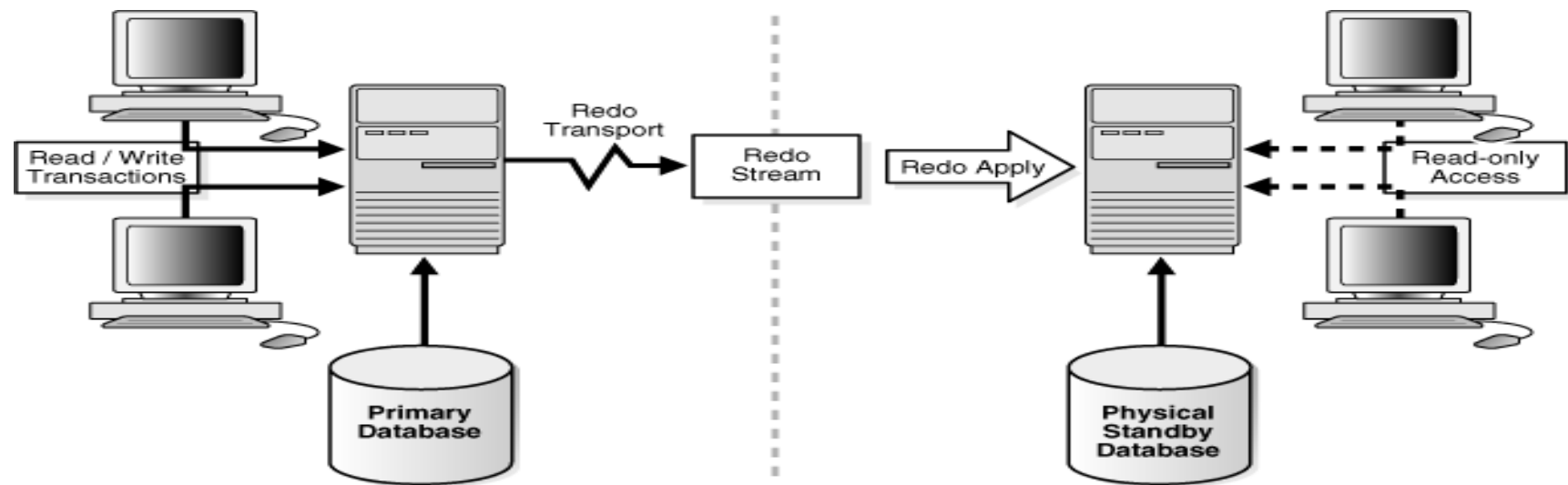


# Outline

- CERN and Oracle
- **Architecture**
- Use Cases for ADG@CERN
- Our experience with ADG and lessons learned
- Conclusions



# Active Data Guard – the Basics



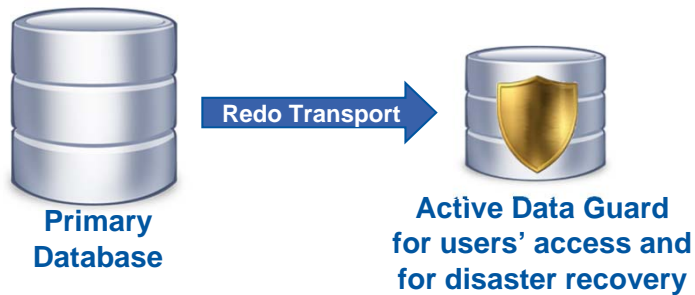
- ADG enables **read only access** to the physical standby

# Active Data Guard @CERN

- **Replication**
  - Inside CERN from online (data acquisition) to offline (analysis)
  - Replication to remote sites (being considered)
- **Load Balancing**
  - Offloading queries
  - Offloading backups
- **Features available also with Data Guard**
  - Disaster recovery
  - Duplication of databases
  - Others

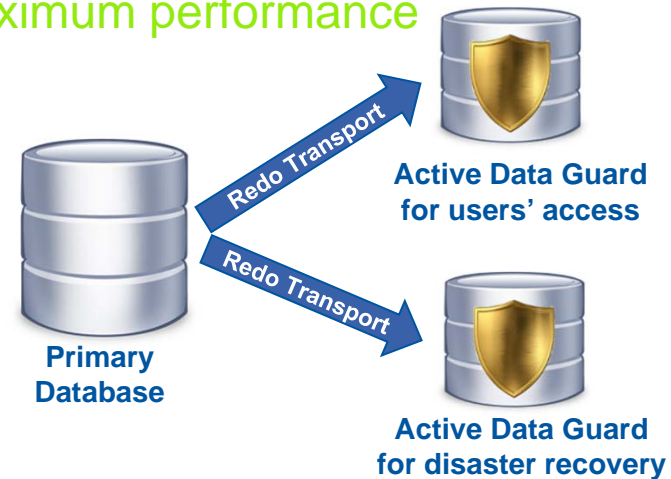
# Architecture We Use

Maximum performance



1. Low load ADG

Maximum performance



2. Busy & critical ADG

```
LOG_ARCHIVE_DEST_X='SERVICE=<tns_alias> OPTIONAL
ASYNC NOAFFIRM VALID FOR=(ONLINE_LOGFILES,PRIMARY_ROLE)
DB_UNIQUE_NAME=<standby_db_unique_name>'
```

# Deployment Model

- Production RAC systems
  - Number of nodes: 2 - 5
  - NAS storage with SSD cache
- (Active) Data Guard RAC
  - Number of nodes: 2 - 4
  - Sized for average production load
  - ASM on commodity HW, recycle 'old production'

# Outline

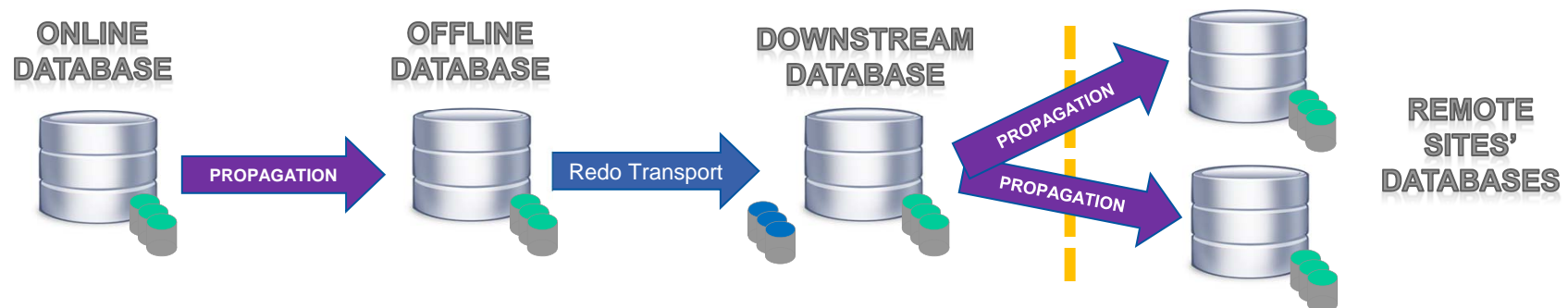
- CERN and Oracle
- Architecture
- **Use Cases for ADG@CERN**
- Our experience with ADG and lessons learned
- Conclusions





# Replication Use Cases

- Number of replication use cases
  - Inside CERN
  - Across the Grid
- So far handled by Oracle Streams
  - **Logical replication** with messages called LCRs (Logical Change Record)
  - Service started with 10gR1, now 11gR2



# ADG vs. Streams for Our Use Cases

## Active Data Guard

- Block level replication
  - More robust & Lower latency
  - Full database replica
- Less maintenance effort 😊
- More use cases than just replication 😊
- Complex replication cases not covered 😞
- Read-only replica 😞

## Streams

- SQL based replication
  - More flexible
  - Data selectivity
- Replica is accessible for read-write load 😊
- Unsupported data types 😞
- Throughput limitations 😞
- Can be complicated to recover 😞

# Offloading Queries to ADG

- **Workload distribution**
  - Transactional workload runs on primary
  - Read-only workload can be moved to ADG
  - Read-mostly workload
    - DMLs can be redirected to remote database with a dblink
- **Examples of workload on our ADGs:**
  - Ad-hoc queries, analytics and long-running reports, **parallel queries**, unpredictable workload and test queries

# Offloading Backups to ADG

- Significantly **reduces load** on primary
  - Removes sequential I/O of full backup
- ADG has great improvement for VLDBs
  - allows usage of block change tracking for fast incremental backups

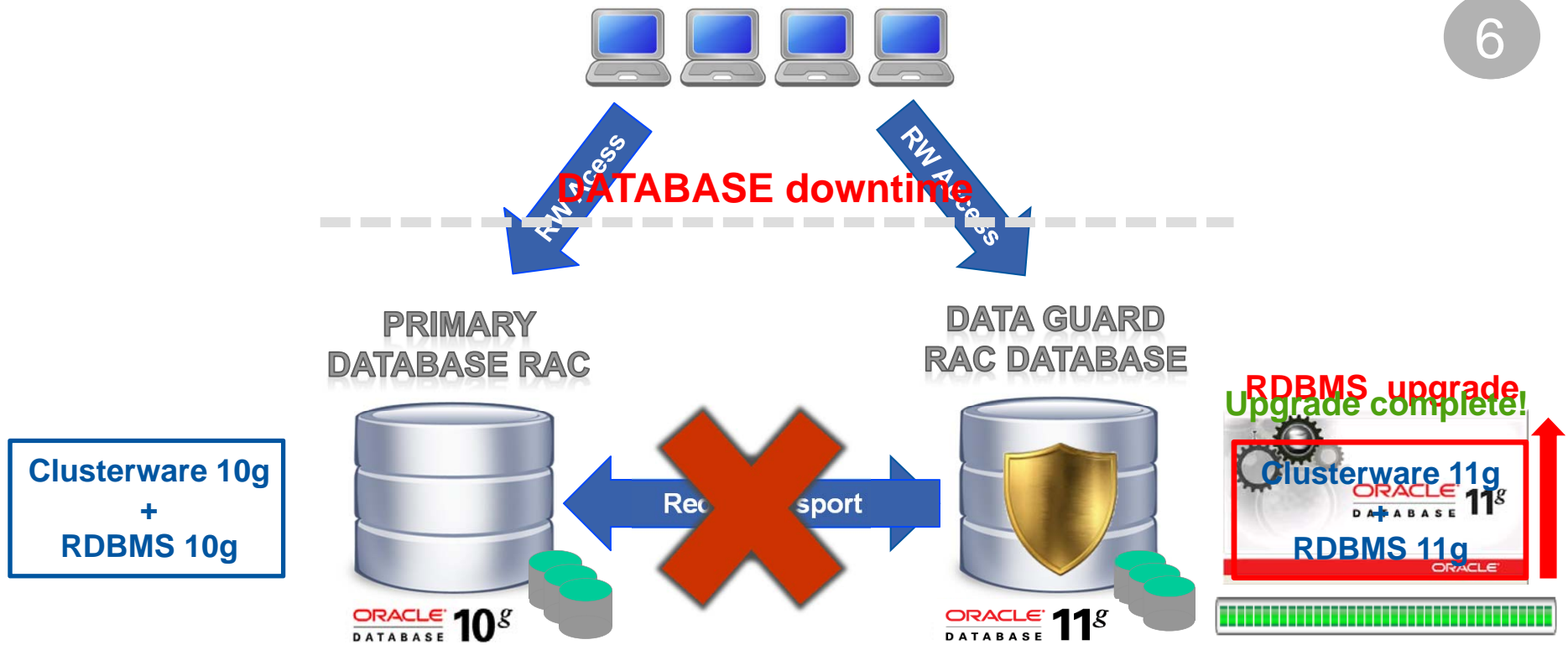
# Disaster Recovery

- We have been using it since a few years
  - Switchover/failover is our **first line of defense**
  - Saved the day already for production services
- Disaster recovery site at ~10 km from our datacenter
- In the future remote site in Hungary

\* Active Data Guard option not required

# Duplicate for Upgrade

6



# Duplicate for Upgrade - Comments

- Risk is limited 😊
  - Fresh installation of the new clusterware
  - Old system stays untouched
  - Allows full upgrade test
  - Allows stress testing of new system
- Downtime is limited 😊
  - ~ 1h for RDBMS upgrade
- Additional hardware is required 😞
  - Only for the time of the upgrade

\* Active Data Guard option not required

# More Use Cases

- Load testing
  - Real application testing (RAT)
- Recover objects against logical corruption
  - Human errors
- Leverage **flashback logs**
  - Additional writes may have the negative impact on production database
- Data lifecycle activities



# Snapshot Standby

- Facilitates opening standby in **read-write** mode
  - Number of steps is limited to two

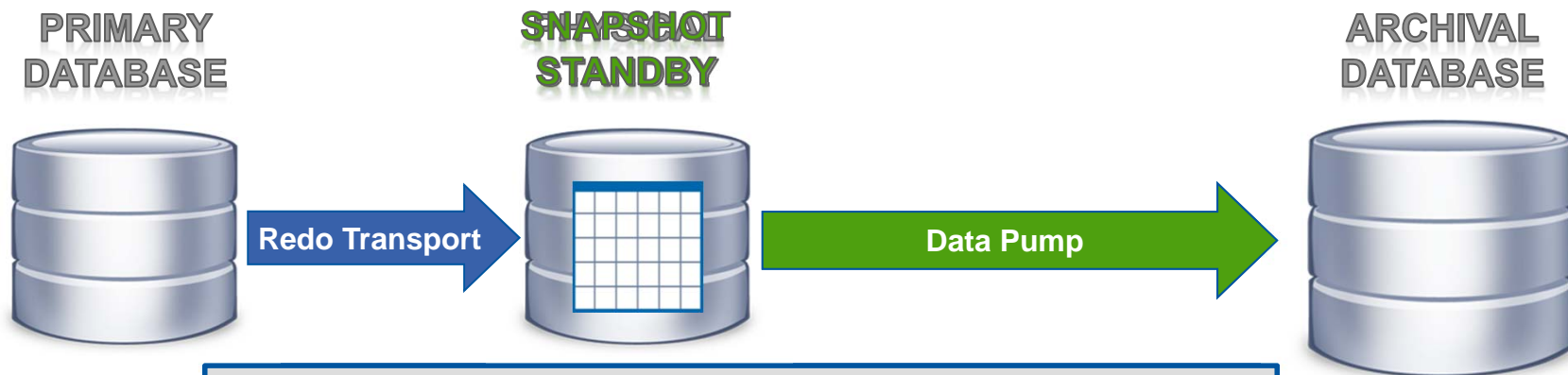
```
SQL> ALTER DATABASE CONVERT TO SNAPSHOT STANDBY;
```

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

- Restore point created internally by Oracle
- Primary database is always protected
  - Redo is being received when standby is opened

# Data Lifecycle Activities

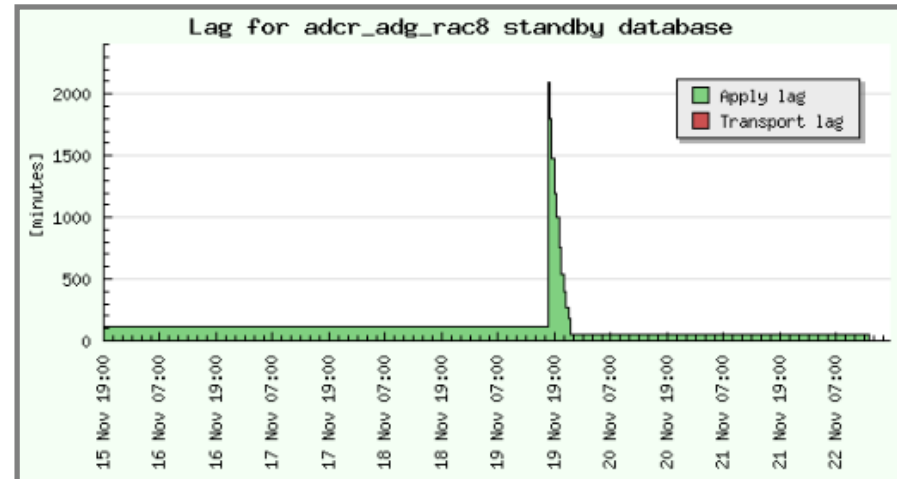
- Transport big data volumes from busy production DBs
  - Transportable Tablespaces (TTS), Data Pump
  - Challenge: TTS requires read only tablespaces



```
SQL> ALTER TABLESPACE data_2011q4 READ ONLY;
```

# Custom Monitoring

- Monitoring
  - Availability
  - Latency
  - + Automatic MRP restart
- **Racmon**



Cluster: <u>adcr - RAC for ATLAS ADC</u>	
	DB instances
Availability last 7 days	%
Status	OK

4 Nodes: adcr1  ✓, adcr2

0 Storages:

Sessions (per node): adcr1 (394), adcr2 (319).

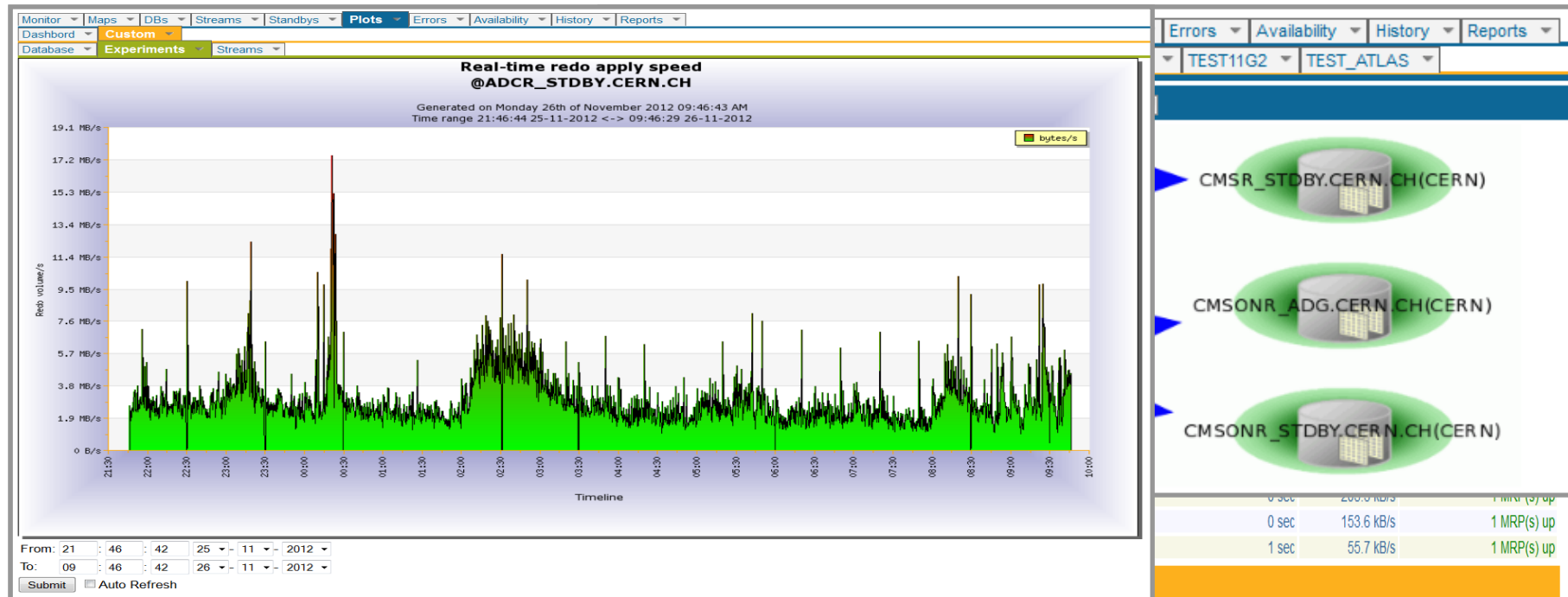
**Subject: RACMon2 Alert: Warning: adcr**  
**Date: November 9, 2012 9:37:25 PM GMT+01:00**  
**To: [redacted]@cern.ch**

---

RAC monitoring errors - cluster adcr (RAC for ATLAS ADC):  
 Managed recovery was down on standby DB at [redacted] - restarted by RACMon!

# Custom Monitoring

- Strmmmon



# Outline

- CERN and Oracle
- Architecture
- Use Cases for ADG@CERN
- **Our experience with ADG and lessons learned**
- Conclusions



# Sharing Production Experience

- ADG in production since Q1 2012
  - Oracle 11.2.0.3, OS: RHEL5 64 bit
- In the following:
  - A few thoughts of how ADG is working in our environment
  - Sharing **experience from operations**

# ADG for Replication

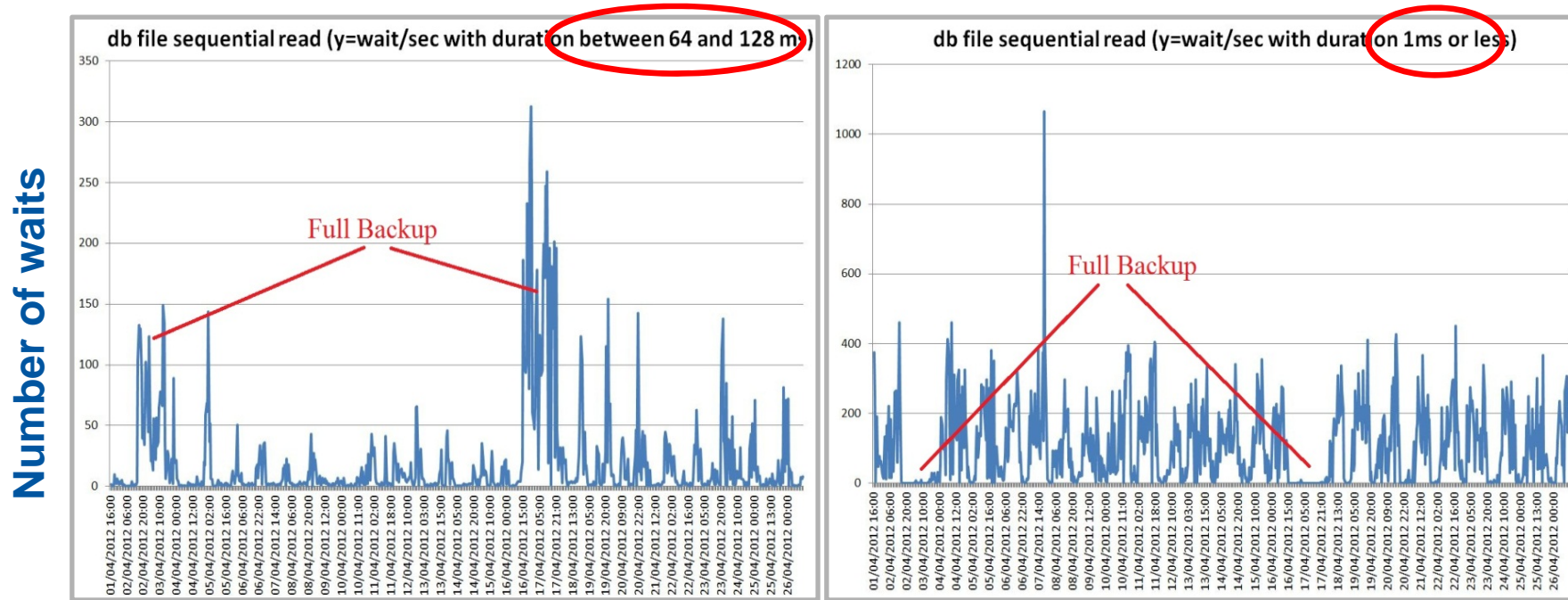
- More robust than Streams
  - **Fewer incidents**, less configuration, less manpower
  - Users like the low **latency**
- More complex replication setups
  - Still on Streams

# Offloading Backups to ADG

- Problem: OLTP production slow during backup
  - Latency for **random IO** is critical to this system
    - degraded by backup sequential IO
  - It's a storage issue:
    - Midrange NAS storage



# Details of Storage Issue on Primary



Very slow reads appear

Reads from SSD cache go to zero

# Custom Backup Implementation

- We have modified our backup implementation and **adapted** it for ADG
  - It was worth the **effort** for us.
  - Many details to deal with.
    - One example: archive log backup and deletion policies
    - Google “Szymon Skorupinski CERN openworld 2012” for details

# Backups with RMAN

- We find backups with **RMAN** still a very good idea
  - Data Guard is not a backup solution
- Automatic restore and recovery system
  - Periodic checks that **backups can be recovered**
- **Block change tracking** on ADG
  - Incremental backups only read changed blocks
  - Highly beneficial for backup performance

# Redo Apply Throughput

- ADG redo apply by MRP
  - Noticeable **improvement in speed** in 11g vs. 10g
  - We see up to ~100 MB/s of redo processed
    - Much higher than typical redo generation in our DBs
- Note on our configuration:
  - Standby logfiles, real time apply + force logging
  - Archiving to ADG using **ASYNC** NOAFFIRM

# Redo Apply Latency

- Latency to ADG normally no more than **1 sec**
  - Apply latency **spikes**: ~20 sec, depends on workload
    - Redo apply slaves wait for **checkpoint completed** (11g)
    - Latency builds up when ADG is **creating big datafiles**
      - One redo apply slave creates file, rest of slaves wait (11g)
    - MRP (recovery) can stop with errors or crashes
  - Possible workarounds:
    - Minimize redo switch by using large redo log files
    - Create datafiles with small initial size and then (auto)extend

# Latency and Applications

- Currently **alarms** to on-call DBAs
  - When latency primary-ADG greater than programmable
- What if we had to guarantee latency
  - Applications maybe would need to be modified to make them **latency-aware**
  - Useful techniques to know
    - `ALTER SESSION SET STANDBY_MAX_DATA_DELAY=<n>;`
    - `ALTER SESSION SYNC WITH PRIMARY; – SYNC only!`

# Administration

- In most cases ADG doesn't require much DBA time after initial setup
- Not much different than handling production DBs
- Although we see **bugs** and **issues** specific to ADG
  - Two examples in following slides

# Failing Queries on ADG

- Sporadic ORA-1555 (snapshot too old) on ADG
  - Normal behaviour if queries need read consistent images outside **undo retention** threshold
  - This **case is anomalous**: affects short queries
  - Under investigation, seems a bug
  - Example:

ORA-01555 caused by SQL statement below  
(SQL ID: ..., **Query Duration=1 sec**, SCN: ...)



# Stuck Recovery and Lost Write

- **Issue:** ORA-600 [3020] “Redo is inconsistent with data block”
  - Critical production with 2 ADGs.
  - Both **ADGs stop** applying redo at the same time... on a Friday night!
  - Primary keeps working
- **Analysis:**
  - Possible cause: lost write on primary
  - Corruption in one index on primary - rebuilt online by DBA on-call
  - The **root cause** seems to be an **Oracle bug** (SR in progress).
- **Impact:**
  - Redo apply on ADG is stuck while issue is being fixed by DBA

# Some Thoughts on Lost Write Issue

- **Complex** recovery case
  - See also note: Resolving ORA-752 or ORA-600 [3020] During Standby Recovery [ID 1265884.1]
- How to be **proactive**?
  - Setting parameter `DB_LOST_WRITE_PROTECT=TYPICAL`
    - It is a warning mechanism not a fix for lost write
    - Impact: just a few percent increase in redo log writes
- Should I **worry** that my primary DB and ADG are not fully synchronized?
  - Not easy to check
  - Note ADG and primary are **not exact binary** copies

# Plans for the Future

- Deploy ADG for **replication to remote sites**
  - Source at CERN, ADG at another grid site
    - ADGs under administration by remote sites' DBAs
- Challenges:
  - Manage heterogeneous environment
  - Redo transport over WAN
  - Maintain a partial standby

# Features Under Investigation

- Data Guard Broker
  - fast-start failover
- Partial standby configuration
- Cascading standby
- Synchronous redo transport
- ADG in 12c

# Outline

- CERN and Oracle
- Architecture
- Use Cases for ADG@CERN
- Our experience with ADG and lessons learned
- **Conclusions**



# Conclusions

- **Active Data Guard** provides **added value** to our database services
  - Almost 1 year of production experience
- In particular ADG has helped us
  - Strengthening our **replication** deployments
  - **Offloading** production workload, including backups
- We find ADG has low maintenance overhead
  - Although requires additional effort in monitoring and **latency** management
  - Also we found a few **bugs** on the way
- We find ADG mature
  - Although we also look forward to bug fixes and **improvements** in next releases
- We are planning to **extend** our usage of ADG

# Acknowledgements

- CERN Database Group
- Experiments community at CERN
- Oracle contacts: Monica Marinucci, Greg Doherty. Larry Carpenter and Michael Smith for discussions



# Thank you!

[Luca.Canali@cern.ch](mailto:Luca.Canali@cern.ch)

[Marcin.Blaszczyk@cern.ch](mailto:Marcin.Blaszczyk@cern.ch)







[www.cern.ch](http://www.cern.ch)