

PEOUG
PERU ORACLE USERS GROUP

Implementando Oracle Streams

Una solución moderna de replicación de datos

Miguel Palacios (miguel.palacios@gbsperu.net)

PEOUG Day - 18 de Noviembre 2009



www.peoug.org

Mi Perfil: Miguel Palacios

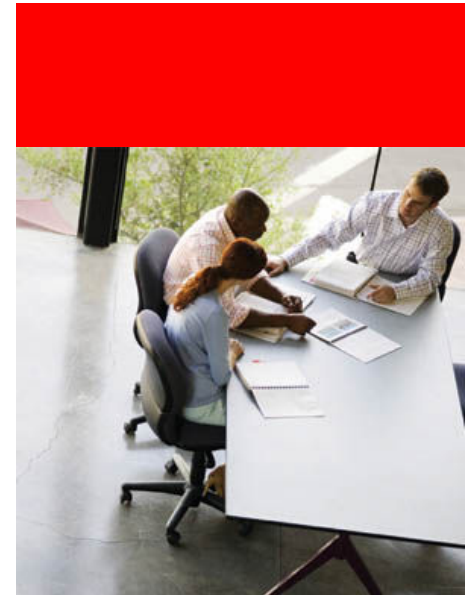
- Fundador y CEO de Global Business Solutions Perú (GBSPERU.NET)
- Líder de PERU Oracle Users Group (PEOUG.ORG)
- Certificación OCP9i, OCP10g, 10g RAC Expert
- Experiencia Oracle:
 - Arquitecto de soluciones con foco en infraestructura
 - 7+ años Instructor certificado de Oracle University
 - Oracle Database Beta Tester desde 2006
 - 10+ años de Advanced Database Administration & Performance Tuning
 - Oracle Backup & Recovery
 - Oracle ASM: 10g R2, 11g R1, 11g R2
 - Oracle Data Guard: Physical and Logical 9iR2, 10gR2, 11gR1 (Active Data Guard). Despliegues de “basic standby” sobre Standard Edition 8i, 9iR2, 10g R2.
 - Oracle RAC: 9iR2, 10GR2, 11gR1 sobre HP-UX Itanium, Linux Itanium, Linux x86_64, Solaris, AIX (LPAR and VIO Server), Windows. Single Data Center & RAC Extendido
 - Oracle Streams: 9i R2, 10g R2 (Database Level y Schema Level: Bi-Directional y Uni-Directional). Upstream & Downstream Capture
 - Oracle VM: Despliegues de Oracle RAC 10gR2, Grid Control 10g, Oracle Enterprise Linux
 - Oracle E-Business Suite 11i: Advanced Performance and Troubleshooting



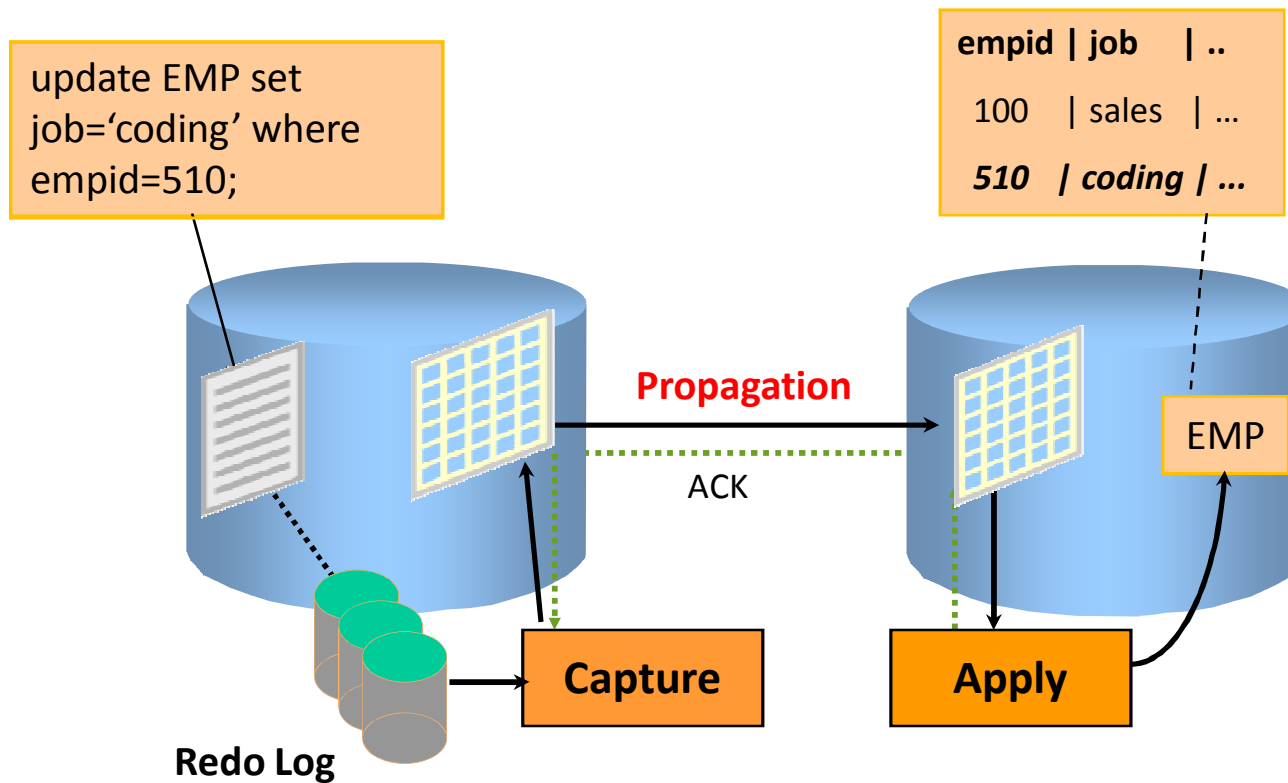
Oracle Streams

- Simple solution for information sharing
- Heterogeneous Replication
- Provides
 - Uniquely flexible information sharing
 - Active/Active replication
 - Real-time ETL for data warehouses
 - Availability during database migration
 - Availability during application upgrade
 - Message queuing
 - Event management and notification

Oracle Streams Architecture



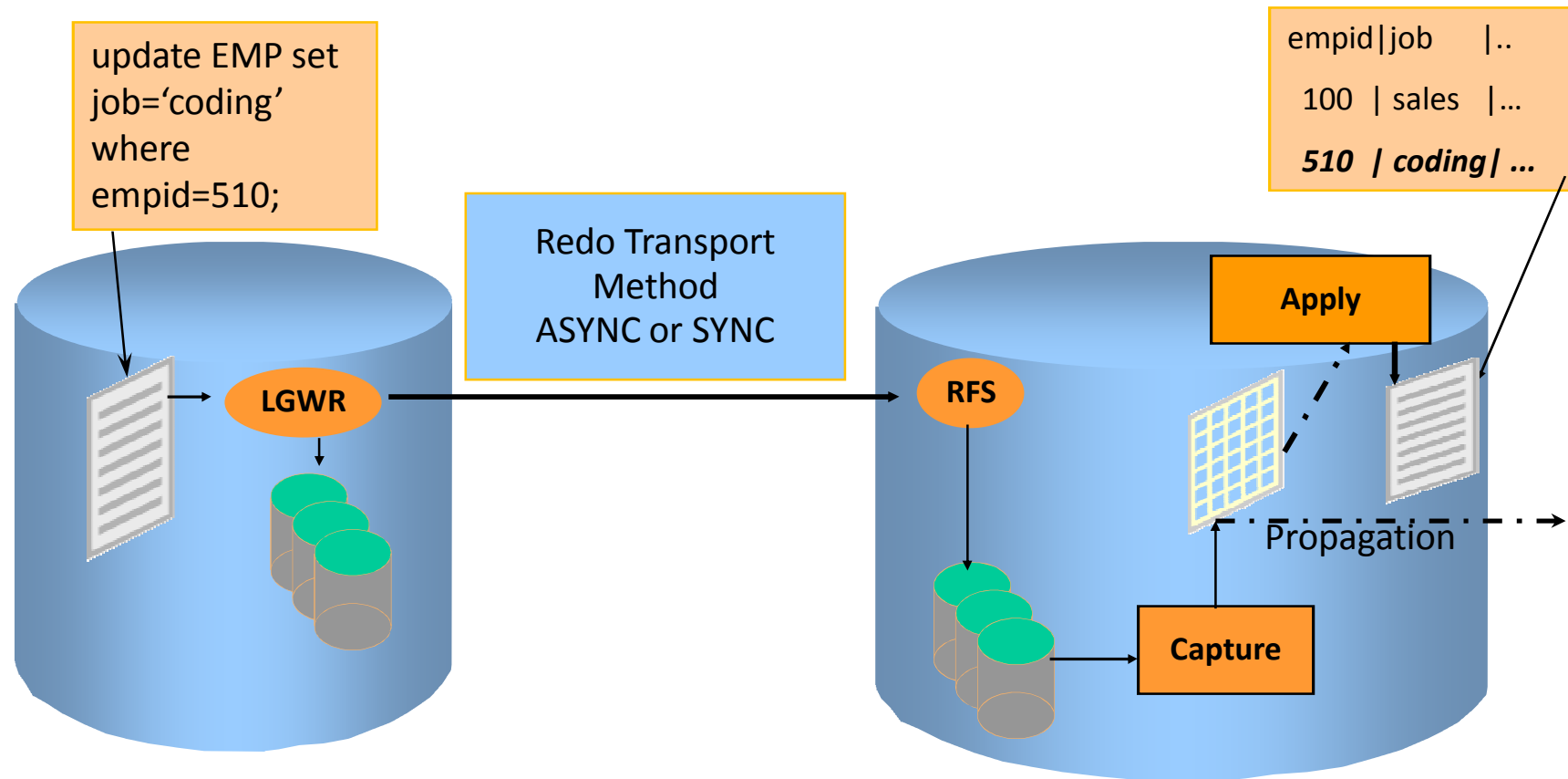
Oracle Streams Capture and Apply *Replication Configuration*



Local (Upstream) Capture

If subset of data, network bandwidth requirements reduced significantly

Oracle Streams Downstream Capture

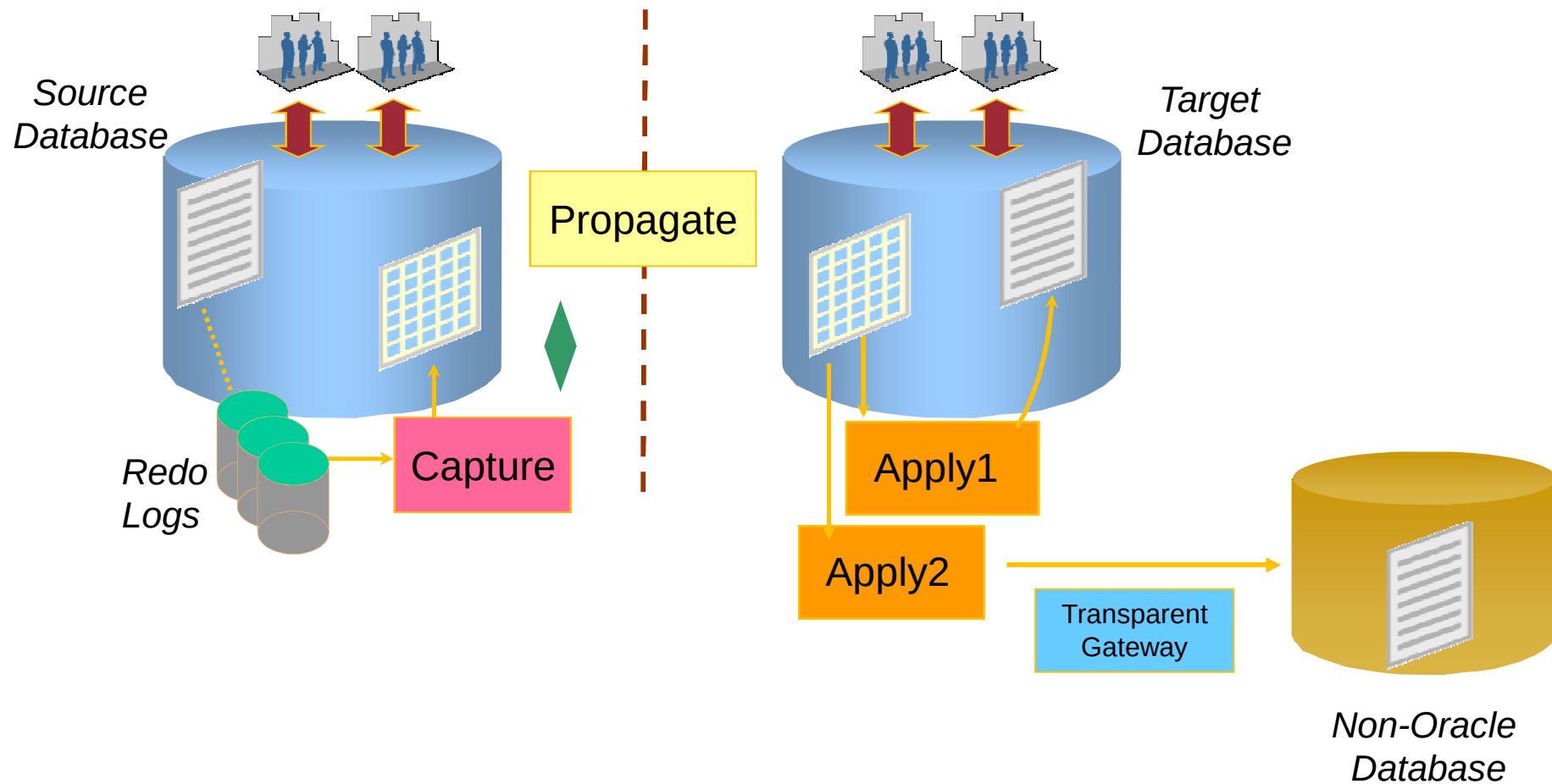


Remote (Downstream) Capture

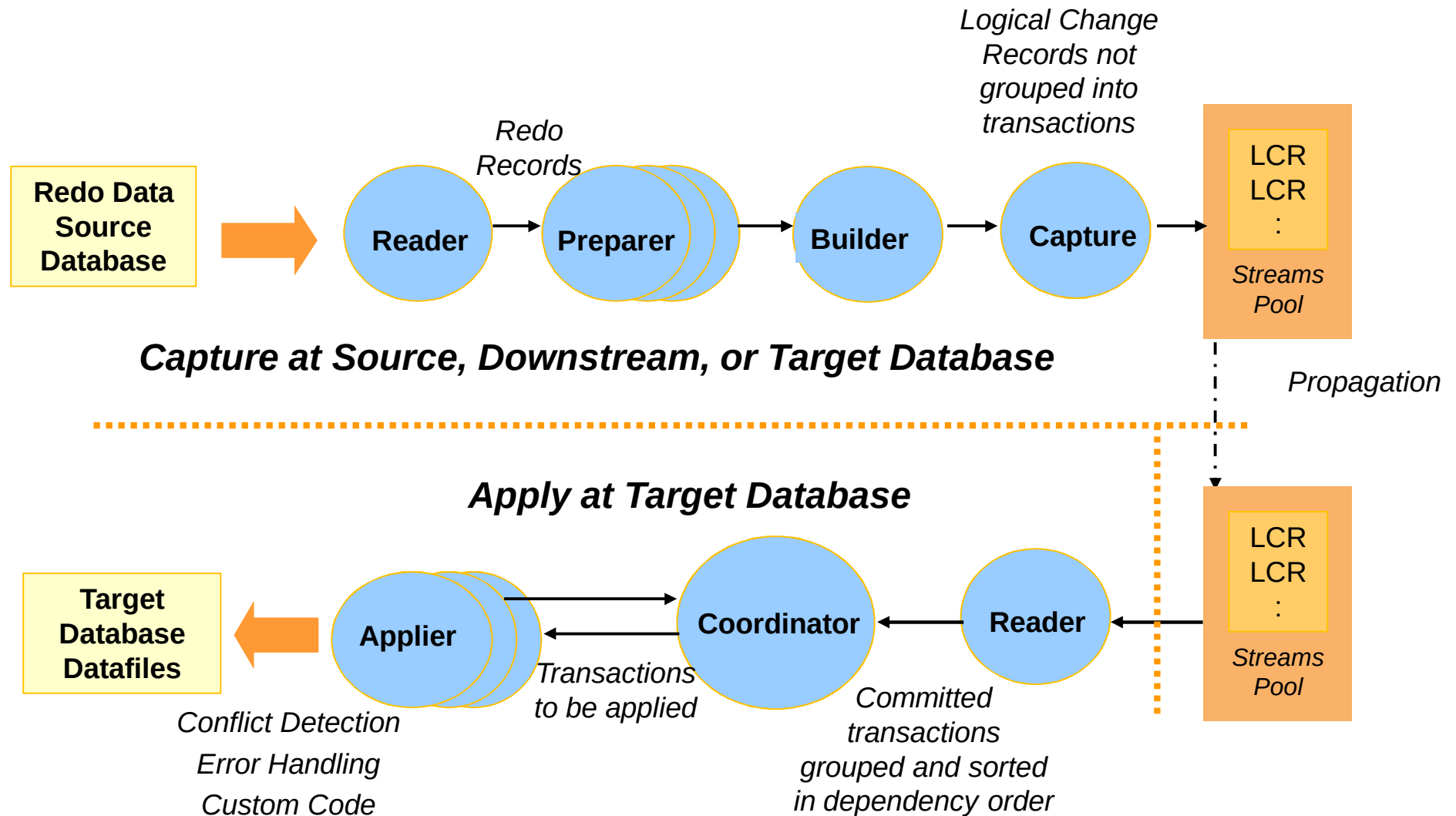
Offload capture from production database

Extended Streams Architecture

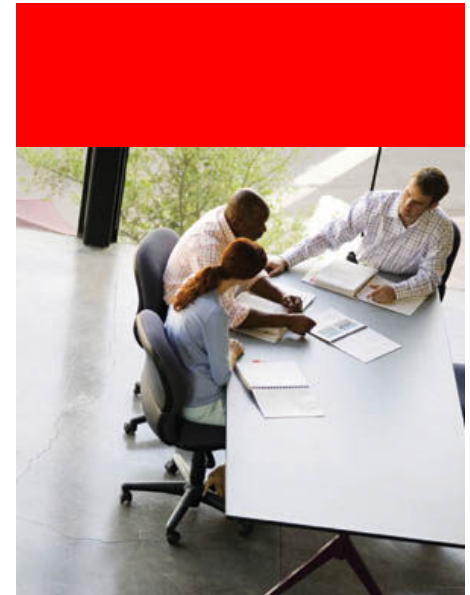
Powerful Solution for Information Sharing



Oracle Streams Process Architecture



Oracle Streams Configuration



Oracle Streams Configuration: Example

- One SOURCE with one TARGET
- One-way replication at SCHEMA level (HR)
- DML and DDL Capture on TARGET (Local Capture)
- SOURCE Global Name: S1.DOMAIN.COM
- TARGET Global Name: T1.DOMAIN.COM
- SOURCE and TARGET on Enterprise Linux
- SOURCE and TARGET on 10g R2
- SOURCE is on ARCHIVELOG MODE with FORCE LOGGING

Step 0: INIT Parameters in SOURCE and TARGET

- alter system set open_links=4
- alter system set log_parallelism=1
- alter system set logmnr_max_persistent_sessions=4
- alter system set global_names=TRUE
- alter system set parallel_max_servers=5
- alter system set open_links=4
- alter system set aq_tm_processes=2
- alter system set job_queue_processes=12
- alter system set archive_lag_target =3600
- **alter system set "_first_spare_parameter"=50**
- **alter system set "_kgghdsidx_count"=1**
- **alter system set "_job_queue_interval"=1**
- **alter system set streams_pool_size=200M**

For WANs: (SQL*NET parameters)

- Increase SDU (sqlnet.ora, tnsnames.ora, listener.ora)
- Increase send_buf_size, recv_buf_size

Step 1: STRMADMIN User and Privileges

SOURCE and TARGET

```
create tablespace TS_STRADMIN datafile '/path/ts_stradmin01.dbf' size 100M  
autoextend on maxsize unlimited extent management local segment space  
management auto;
```

```
create user STRMADMIN identified by PASSWORD default tablespace TS_STRADMIN;
```

```
GRANT CONNECT, RESOURCE, AQ_ADMINISTRATOR_ROLE, DBA to STRMADMIN;  
GRANT EXECUTE ON DBMS_AQADM TO STRMADMIN;  
GRANT EXECUTE ON DBMS_APPLY_ADM TO STRMADMIN;  
GRANT EXECUTE ON DBMS_CAPTURE_ADM TO STRMADMIN;  
GRANT EXECUTE ON DBMS_PROPAGATION_ADM TO STRMADMIN;  
GRANT EXECUTE ON DBMS_STREAMS TO STRMADMIN;  
GRANT EXECUTE ON DBMS_STREAMS_ADM TO STRMADMIN;  
GRANT EXECUTE ON DBMS_AQ TO STRMADMIN;  
GRANT EXECUTE ON DBMS_FLASHBACK TO STRMADMIN;  
GRANT EXECUTE ON DBMS_RULE_ADM TO STRMADMIN;  
GRANT SELECT ANY DICTIONARY TO STRMADMIN;
```

Step 2: Streams QUEUE in SOURCE and TARGET

```
connect STRMADMIN/PASSWORD
```

```
BEGIN  
DBMS_STREAMS_ADM.SET_UP_QUEUE(  
queue_table => 'STREAMS_QUEUE_TABLE',  
queue_name => 'STREAMS_QUEUE',  
queue_user => 'STRMADMIN');  
END;  
/
```

Step 3: Apply RULES in TARGET

```
connect STRMADMIN/STRMADMIN
```

```
BEGIN
```

```
  DBMS_STREAMS_ADM.ADD_SCHEMA_RULES(  
    schema_name => 'HR',  
    streams_type => 'APPLY',  
    streams_name => 'STRMADMIN_APPLY',  
    queue_name => 'STRMADMIN.STREAMS_QUEUE',  
    include_tagged_lcr => true,  
    include_dml => true,  
    include_ddl => true,  
    source_database => 'S1.DOMAIN.COM');
```

```
END;
```

```
/
```

Step 4: Apply USER in TARGET

```
connect STRMADMIN/STRMADMIN
```

```
BEGIN
```

```
    DBMS_APPLY_ADM.ALTER_APPLY(  
        apply_name => 'STRMADMIN_APPLY',  
        apply_user => 'HR');
```

```
END;
```

```
/
```

```
BEGIN
```

```
    DBMS_APPLY_ADM.SET_PARAMETER(  
        apply_name => 'STRMADMIN_APPLY',  
        parameter  => 'DISABLE_ON_ERROR',  
        value      => 'N' );
```

```
END;
```

```
/
```


Step 5: Logminer Tablespace

```
connect SYS/PASSWORD as SYSDBA
```

```
CREATE TABLESPACE LOGMNRTS DATAFILE  
'/path/logmnrts01.dbf' SIZE 100M AUTOEXTEND ON MAXSIZE  
UNLIMITED;
```

```
BEGIN  
    DBMS_LOGMNR_D.SET_TABLESPACE('LOGMNRTS');  
END;  
/
```

Step 6: Supplemental logging in SOURCE and TARGET

connect SYS/PASSWORD as SYSDBA

- ALTER DATABASE ADD SUPPLEMENTAL LOG DATA;
- ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (PRIMARY KEY, UNIQUE INDEX) COLUMNS;

Recommended at table level:

- ALTER TABLE HR.<TABLE_NAME> ADD SUPPLEMENTAL LOG DATA (PRIMARY KEY) COLUMNS;

or

- ALTER TABLE HR. <TABLE_NAME> ADD SUPPLEMENTAL LOG DATA (ALL) COLUMNS;

Step 7: Database link in SOURCE and TARGET

→ **SOURCE** connect STRMADMIN/STRMADMIN

```
CREATE DATABASE LINK T1.DOMAIN.COM connect to  
STRMADMIN identified by PASSWORD using  
'TARGET_STRING';
```

```
select * from global_name@T1.DOMAIN.COM;
```

→ **TARGET** connect STRMADMIN/STRMADMIN

```
CREATE DATABASE LINK S1.DOMAIN.COM connect to  
STRMADMIN identified by PASSWORD using  
'SOURCE_STRING';
```

```
select * from global_name@S1.DOMAIN.COM;
```

Step 8: Capture and Propagation RULES for HR schema in SOURCE

```
connect STRMADMIN/STRMADMIN
```

```
BEGIN
```

```
  DBMS_STREAMS_ADM.ADD_SCHEMA_RULES(  
    schema_name => 'HR',  
    streams_type => 'CAPTURE',  
    streams_name => 'STREAM_CAPTURE',  
    queue_name => 'STRMADMIN.STREAMS_QUEUE',  
    include_tagged_lcr => true,  
    include_dml => true,  
    include_ddl => true,  
    source_database => 'S1.DOMAIN.COM');
```

```
END;
```

```
/
```

```
BEGIN
```

```
  DBMS_STREAMS_ADM.ADD_SCHEMA_PROPAGATION_RULES(  
    schema_name => 'HR',  
    streams_name => 'STREAM_PROPAGATE',  
    source_queue_name => 'STRMADMIN.STREAMS_QUEUE',  
    destination_queue_name => 'STRMADMIN.STREAMS_QUEUE@T1.DOMAIN.COM',  
    include_tagged_lcr => true,  
    include_dml => true,  
    include_ddl => true,  
    source_database => 'S1.DOMAIN.COM');
```

```
END;
```

```
/
```

Step 9: Export / Import Procedure

→ SOURCE

```
exp USERID=SYSTEM/PASSWORD OWNER=HR FILE=HR.dmp  
LOG=exportHR.log OBJECT_CONSISTENT=Y  
STATISTICS=NONE direct=y recordlength=65535
```

→ TARGET

Create DATABASE structure like TABLESPACE and DATAFILES

```
imp USERID=SYSTEM/PASSWORD full=Y fromuser=HR  
touser=HR CONSTRAINTS=Y FILE=HR.dmp IGNORE=Y  
COMMIT=Y LOG=importHR.log  
STREAMS_INSTANTIATION=Y statistics=none
```

Step 10: Start CAPTURE process at SOURCE and APPLY process at TARGET

→ SOURCE

```
connect STRMADMIN/STRMADMIN
```

```
begin
    DBMS_CAPTURE_ADM.START_CAPTURE(capture_name =>
        'STREAM_CAPTURE');
end;
/
```

→ TARGET

```
connect STRMADMIN/STRMADMIN
```

```
BEGIN
    DBMS_APPLY_ADM.START_APPLY(apply_name =>
        'STRMADMIN_APPLY');
END;
/
```

Oracle Streams: Use Cases

- Offload transactional database using a replica database for adhoc queries and reporting.
- Maintain updated QA or DEV Databases.
- Upgrade database version with minimal downtime, for example 9iR2 to 10g R2 or 10gR2 to 11gR1 using STREAMS capture process.
- Using Streams as part of the Oracle Disaster Recovery solutions, for example SOURCE on AIX with a streams replica on Linux (*DELETES could be a problem!*)
- Distributed databases, for example one local master database with several remote (replica) databases. Hub-and-Spoke Configurations.
- Streams replication using Enterprise Edition as SOURCE database with Standard Edition as TARGET database (one-way streams replication)
- Oracle 11g SE -> SE synchronous replication



PEOUG

PERU ORACLE USERS GROUP

www.peoug.org