
Oracle Application Continuity (AC)

Author	김성용
Creation Date	2019-08-19
Last Updated	
Version	1.0
Copyright(C) GoodusData Inc. All Rights Reserved	

Version	변경일자	변경자(작성자)	주요내용
1	2019-08-19	김성용	초안작성

Contents

1. Application Continuity(AC) 란?	3
1.1. Key Concepts for Application Continuity	3
2. 동작방식	4
3. Application Continuity 요구사항.....	7
4. Configuration Application Continuity – 구성	8
4.1. Application Continuity for Oracle Single Instance.....	8
4.2. Session TEST	13
4.3. NOREPLAY Option	15
4.4. Application Continuity for Oracle Rac	18
4.5. NoReplay Option	23
5. Potential Side Effects of Application Continuity	24
6. Application Continuity 제약사항	25
7. References.....	25

1. Application Continuity(AC) 란?

Oracle Database 12 c Release 1 (12.1.0.1)에 도입 된 Application Continuity 는 Oracle 데이터베이스를 사용하는 System 및 Application 의 내적 결함을 강화 시키는 기능입니다.

Application Continuity 는 Session 의 장애(Session 의 중지)를 빠르고 연속성 있게 복구 해주는 기능입니다. replay 이후 application 은 session 이 중지되어도 작업을 이어서 수행합니다.

1.1. Key Concepts for Application Continuity

1. Request

Request 는 Application 에서 보내온 SQL, PL/SQL 이나 다른 Database Call 과 같은 작업의 단위 입니다. Recoverable Error 에서 AC 는 세션의 상태를 다시 설정하고 요청을 반복합니다.

2. Recoverable Error

Recoverable Error 는 실행중인 Application Session 과 관계없는 외부 시스템 오류로 인해 발생하는 오류를 뜻합니다. AC 는 Session 을 다시 설정하고 Recoverable Error 에 대해 작업을 다시 요청합니다.

3. Commit Outcome

트랜잭션은 트랜잭션 테이블에서 항목을 업데이트하여 커밋됩니다. Oracle Database 는이 업데이트에 해당하는 리두 로그 레코드를 생성하고이 리두 로그 레코드를 작성합니다. 이 리두 로그 레코드가 디스크의 리두 로그에 기록되면 트랜잭션은 데이터베이스에서 커밋 된 것으로 간주됩니다. 클라이언트 관점에서 볼 때, 리두가 작성된 후 생성 된 Oracle 메시지 (Commit Outcome) 가 클라이언트에 의해 수신 되면 트랜잭션은 커밋 된 것으로 간주 됩니다.

4. Mutable Objects

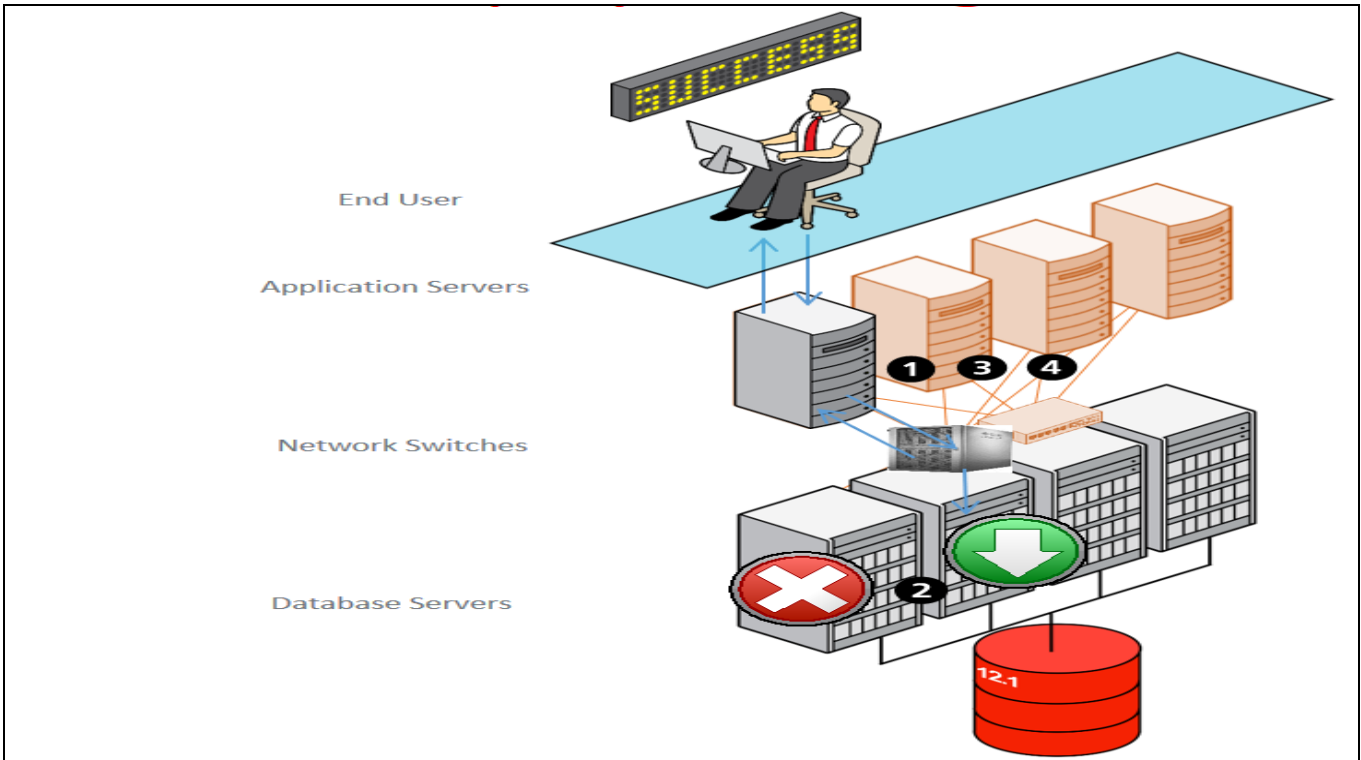
Mutable Objects 는 호출될 때 마다 새로운 값이 할당 될 수 있는 함수입니다.

Sequence.,nextval, SYSDATE 가 Key value 로 사용됩니다. AC 에서는 이러한 바인드 변수에 대한 일관성을 제공합니다. Replay 시에 Sequence.netxtval , SYSDATE , SYSTIMESTAMP 및 SYSGUID 등의 Original value 는 이전 호출의 값을 갖게 됩니다.

5. Session state consistency

Commit 이 실행 된 후 트랜잭션에서 상태가 변경된 경우 세션이 유실되면 해당 상태를 다시 설정하기 위해 트랜잭션을 재생 할 수 없습니다. Application Continuity 를 구성 할 때 Application 은 초기설정 후의 세션 상태가 정적인지 동적인지 Commit 에 따라 요청 내에서 작업을 계속하는 것이 올바른지 여부에 따라 분류됩니다.

2. 동작방식

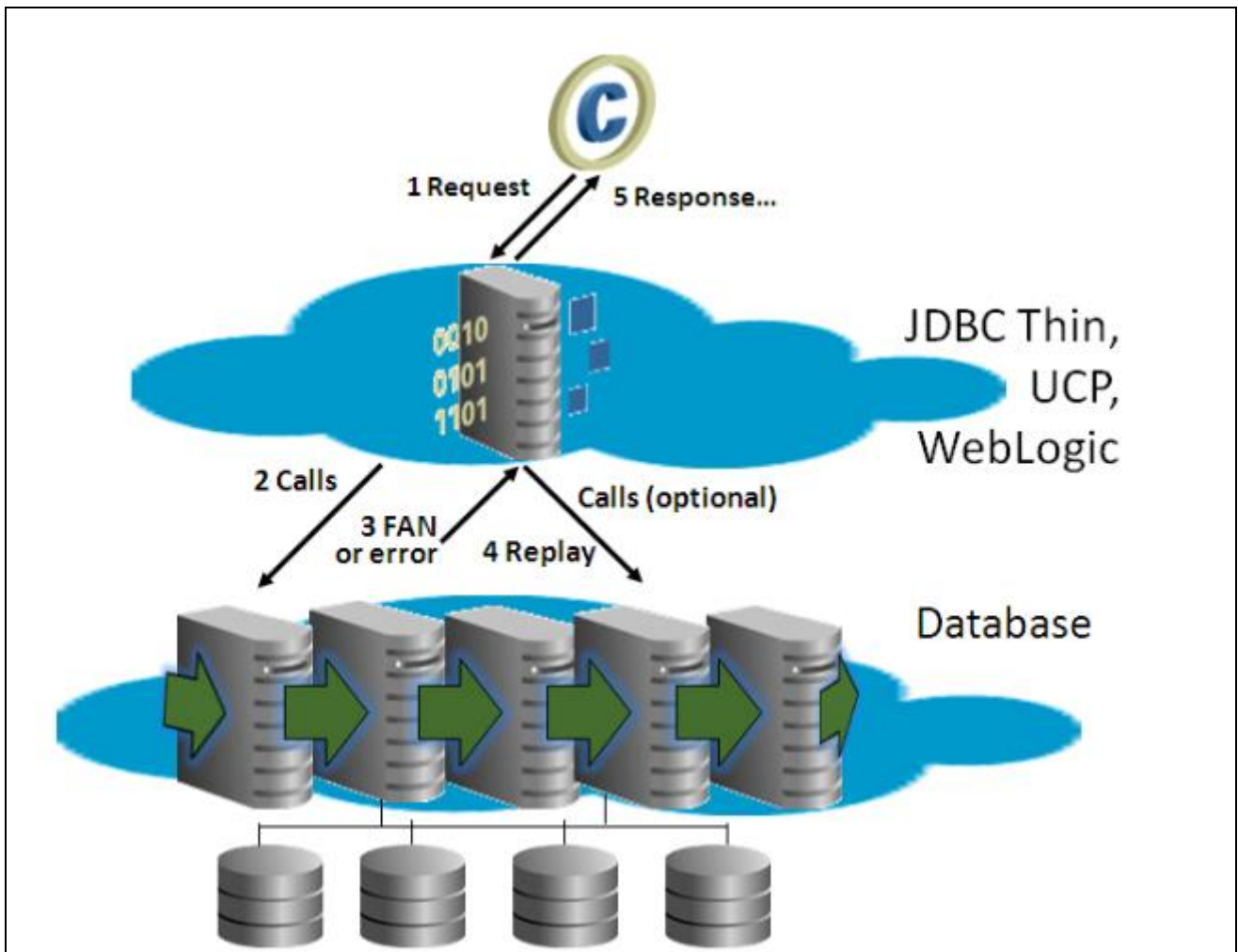


1. 사용자가 Application 을 통해 사용자 트랜잭션이 응용 프로그램 인프라에 도착합니다. 응용 프로그램 계층을 통과하여 데이터베이스 트랜잭션이 생성됩니다.

2.데이터베이스를 호스팅하는 인프라는 트랜잭션이 데이터베이스에 커밋되기 직전에 실패합니다.

3. JDBC Thine Driver 는 세션의 오류를 감지하고 트랜잭션을 재생성 해야하는 경우 Application Continuity 가 작업을 복구합니다.

4. 사용자는 요청이 성공적으로 수행되었다는 결과를 받습니다.



1. 클라이언트 Application 은 JDBC Thin Driver 를 사용하여 Middle-Tier(JDBC Thin driver, Universal Connection Pool, or WebLogic Server or third party pool) 또는 Direct 로 데이터베이스로 의 연결을 요청하게 됩니다.

2. JDBC replay driver 는 각 요청에서 호출을 발생시킵니다

3. FAN 은 Interrupt 또는 Recoverable Error 가 발생합니다. 그 후 FAN / FCW 는 물리적 세션을 중단시킵니다.

4. Application Continuity 는 Replay 를 시도 하고 다음 단계를 수행합니다.

1. Replay 중 또는 Replay 후 오류가 발생하면 사용 불가능한 물리적 세션을 새로운 세션으로 교체하고 FAN 을 다시 연결합니다.

2. 트랜잭션이 열려있는 경우 Transaction Guard 를 사용하여 Replay 을 준비합니다.

3. 라벨링 콜백을 사용하여 콜백하거나 초기 상태에 대한 콜백을 다시 연결이 가능합니다.

4. 데이터베이스 세션을 다시 연결하고 트랜잭션 및 비 트랜잭션 상태를 복구하며 클라이언트 드라이버가 보는 데이터 및 메시지가 클라이언트가 보고 결정을 내리는 데 사용한 데이터 및 메시지와 동일한 지 각 단계에서 유효성을 검증합니다.

5. Replay 을 종료하고 런타임 모드로 돌아갑니다.

6. 마지막으로 대기하고 있던 Call 을 수행합니다.

5. Response 가 Application 으로 Return 됩니다.

3. Application Continuity 요구사항.

- AC 는 다음의 Oracle technologies 에서 사용 가능합니다.
 - JDBC Thin Oracle replay driver
 - Universal Connection Pool
 - WebLogic Server (WebLogic Server 12.1.2 이상)
- AC 를 사용하기 위해서는 Oracle JDBC concrete classes 를 사용하는지 확인해야 합니다. 사용되지 않는 concrete classes 는 반드시 Replaced 되어야 합니다. (Note 1364193.1 참조)

- CPU

Application Continuity 는 Client 와 Server 측에서 관리 되며 작동 하려면 추가적인 CPU 의 여유가 필요합니다.

- Client 측면에서 CPU 는 proxy objects 를 빌드하고 garbage collection(GC) 하기 위해 필요합니다.
- Server 측면에서 CPU 는 유효성 검사를 위해 CPU 가 필요합니다.

- Memory

요청이 끝날 때 까지 호출이 유지되므로 replay driver 가 구동되기 위해서는 Memory 가 필요합니다. 요청이 끝나면 호출은 garbage collection 에 의해서 Release 됩니다.

- Oracle WebLogic Server 나 the Universal Connection Pool 를 사용 할 경우 connection labeling 사용을 권장합니다.
- 12c Release 1 (12.1.0.1) or later 이상 버전에서 사용이 가능합니다.
- Universal Connection Pool 12.1 (or later) or WebLogic Server 12.1.2 (or later) configured with the JDBC Replay data source
- Connect 을 하기 위해 Service Name 을 사용해야 합니다.(SID 혹은 Instance Name 사용 불가)

4. Configuration Application Continuity – 구성

4.1. Application Contunuity for Oracle Single Instance

- Single Instance 일 경우 **DBMS_SERVICE Package** 를 사용하여 구성이 가능합니다.
- DBMS_SERVICE
 - Create_Service Procedure

Parameter	Description
Service_name	Service name, limited to 64 characters
Network_name	클라이언트 연결을 위해 SQLNet 에 사용되는 Service 의 network name
Goal	Workload Management Goal directive for the service. (Default Null) <ul style="list-style-type: none"> ▪ DBMS_SERVICE>GOAL_SERVICE_TIME ▪ DBMS_SERVICE.GOAL_THROUGHPUT ▪ DBMS_SERVICE.GOAL_NONE
Dtp	X/Open DTP(Open Distributed Transaction Processing) 또는 모든 분산 트랜잭션 (특히 XA)을위한 서비스임을 선언합니다. (Default Null)
Aq_ha_notifications	OCI / OCCI / ODP 에 FAN(Fast Application)을 사용할지 여부를 결정합니다. Oracle 12c 에서 FAN 은 Oracle Notification Service(ONS)를 사용합니다. (Default Null)
Failover_method	Application Continuity 와 TAF Service 의 Failover TYPE. Failover_type 이 TRANSACTION 으로 설정되어 있으면 Failover_method 는 자동으로 Commit_Outcome 값을 True 로 설정합니다.
Failover_type	Application Continuity 와 TAF Service 의 Failover TYPE.
Failover_retries	Application Continuity 와 TAF Service 의 Connection 재시도 횟수.
Failover_delay	Application Continuity 와 TAF Service 의 reconnection 사이의 시간, Application Continuity 의 기본값은 10 sec.
Commit_outcome	Commit 이 실행된 후, Transaction 결과에 Access 할 수 있는지 여부를 결정 합니다.
Retention_timeout	Commit_outcome 과 함께 사용, Commit_outcome 이 유지되는 시간(초)을 결정 합니다. 기본값 24 시간(86400), 최대 30 일(2592000) 입니다.
Sql_transaction_name	SQL Translation Unit 의 이름.
Clb_goal	Connection Load Balancing 에 사용되는 Method.

- Delete_service Procedure
- Data Dictionary 에 있는 Service 를 삭제할 때 수행 되는 Procedure

Parameter	Description
Service_name	Service name, limited to 64 characters

- Disconnect_session Procedure

- Instance 에서 해당 Service 와 연결을 끊어주는 Procedure

Parameter	Description
Service_name	Service name, limited to 64 characters
Disconnect_option	<ul style="list-style-type: none"> POST_TRANSACTION : 현재 트랜잭션 Commit 또는 Rollback 후 세션 끊기 Immediate : 즉시 Session 연결을 끊기. NOREPLAY : Session 연결이 즉시 끊어지고 Application Continuity 에 Flagged 되지 않는다.

- Start_Service Procedure

- Service 를 시작하는 Procedure

Parameter	Description
Service_name	Service name, limited to 64 characters
Instance_name	Service 를 중지 할 Instance_name. Default NULL

- Stop_Service Procedure

- Service 를 중지하는 Procedure

Parameter	Description
Service_name	Service name, limited to 64 characters
Instance_name	Service 를 중지 할 Instance_name. Default NULL

- Service 생성.

##Create Service

```
SQL> exec dbms_service.create_service('ACTEST','ACTEST');
```

PL/SQL procedure successfully completed.

##Modify Service

```
SQL> declare
params dbms_service.svc_parameter_array;
begin
```

```
params('FAILOVER_TYPE'):= 'TRANSACTION';
params('REPLAY_INITIATION_TIMEOUT'):=1800;
params('RETENTION_TIMEOUT'):=86400;
params('FAILOVER_DELAY'):=10;
params('FAILOVER_RETRIES'):=30;
params('commit_outcome'):= 'true';
params('aq_ha_notifications'):= 'true';
dbms_service.modify_service('ACTEST',params);
end;
/
```

PL/SQL procedure successfully completed.

##Start Service

```
SQL> exec DBMS_SERVICE.START_SERVICE('ACTEST');
```

PL/SQL procedure successfully completed.

- JDBC 연동

```
import java.sql.*;
import oracle.jdbc.*;
public class AcTest
{
public static void main(String[] args) throws SQLException,java.lang.InterruptedException
{
oracle.jdbc.replay.OracleDataSource  AcDatasource =
oracle.jdbc.replay.OracleDataSourceFactory.getOracleDataSource();
AcDatasource.setURL("jdbc:oracle:thin:@ 172.40.40.55:1523/ACTEST");
AcDatasource.setUser("ACTEST");
AcDatasource.setPassword("welcome1");

Connection conn = AcDatasource.getConnection();

conn.setAutoCommit(false);

PreparedStatement stmt = conn.prepareStatement("select instance_name from v$instance");
```

```

ResultSet rset = stmt.executeQuery();
while (rset.next())
{
System.out.println("You are Connected to RAC Instance - "+ rset.getString(1));
}

Thread.currentThread().sleep(60000);

((oracle.jdbc.replay.ReplayableConnection)conn).beginRequest();

PreparedStatement stmt1 = conn.prepareStatement("select instance_name from v$instance");
ResultSet rset1 = stmt1.executeQuery();
while (rset1.next())
{
System.out.println("After Replay Connected to RAC Instance - "+rset1.getString(1));
}

rset.close();
stmt.close();
rset1.close();

stmt1.close();

conn.close();

((oracle.jdbc.replay.ReplayableConnection)conn).endRequest();
}
}

```

- Source File Comfile

```

[GOODUSDATA:TESTDB:/home/oracle/JAVA]/u01/app/12c/jdk/bin/javac -classpath
/u01/app/12c/jdbc/lib/ojdbc8.jar AcTest.java
[GOODUSDATA:TESTDB:/home/oracle/JAVA]ls -lrt
total 8
-rw-r--r--. 1 oracle dba 1567 Jan  8 12:56 AcTest.java
-rw-r--r--. 1 oracle dba 2028 Jan  8 13:55 AcTest.class

```

```
[GOODUSDATA:TESTDB:/home/oracle/JAVA]/u01/app/12c/jdk/bin/java -classpath
/u01/app/12c/jdbc/lib/ojdbc8.jar:. AcTest
```

You are Connected to Single Instance - TESTDB

After Replay Connected to Single Instance – TESTDB

➔ 중간에 다른 Session 으로 접속해서 Kill session 을 수행해도 해당 Session 을 Replay 하여 수행됩니다.

```
SQL> @sess
```

SID	SERIAL#	USERNAME	MACHINE	MODULE
1347	28097	ACTEST	GOODUSDATA	JDBC Thin Client

STATUS

INACTIVE

```
SQL> alter system kill session '1347,28097';
```

System altered.

```
SQL> @sess
```

SID	SERIAL#	USERNAME	MACHINE	MODULE
1347	28097	ACTEST	GOODUSDATA	

STATUS

KILLED

4.2. Session TEST

- JDBC Thine Client 를 통하여 DML(Insert)를 발생시켜 Application Continuity 작동을 Test.
 - Connection 후 ACTEST schema 의 INFO Table 에 Insert 후 60 초간 Sleep 상태로 있다가 Disconnection 전에 Insert 를 실행하는 TEST 문.

```
public class AcTest {
    public static void main(String[] args) throws SQLException, java.lang.InterruptedExcepion {
        oracle.jdbc.replay.OracleDataSource AcDatasource =
oracle.jdbc.replay.OracleDataSourceFactory.getOracleDataSource();
        AcDatasource.setURL("jdbc:oracle:thin:@172.40.40.55:1523/ACTEST");
        AcDatasource.setUser("ACTEST");
        AcDatasource.setPassword("welcome1");

        Connection conn = AcDatasource.getConnection();

        conn.setAutoCommit(false);
            String sqlstr = "insert into ACTEST.INFO values";
            sqlstr = sqlstr
                + "(SYSDATE,"
                + "ACTEST.INFO_ID_SEQ.nextval,"
                + "'Connected')";
        PreparedStatement stmt = conn.prepareStatement(sqlstr);
        ResultSet rset = stmt.executeQuery();

        while (rset.next()) {
            stmt = conn.prepareStatement("commit");
            rset = stmt.executeQuery();
        }

        Thread.currentThread().sleep(60000);

        ((oracle.jdbc.replay.ReplayableConnection) conn).beginRequest();

            String sqlstr1 = "insert into ACTEST.INFO values";
            sqlstr1 = sqlstr1
                + "(SYSDATE,"
                + "ACTEST.INFO_ID_SEQ.nextval,"
                + "'Disconnected')";
        stmt = conn.prepareStatement(sqlstr1);
```

```

        rset = stmt.executeQuery();

        while (rset.next()) {
            stmt = conn.prepareStatement("commit");
            rset = stmt.executeQuery();

        }

        rset.close();
        stmt.close();

        conn.close();

        ((oracle.jdbc.replay.ReplayableConnection) conn).endRequest();
    }
}

```

- 스크립트 수행

```

[GOODUSDATA:TESTDB:/home/oracle/JAVA]date
Sun Jan 19 18:47:32 KST 2020
[GOODUSDATA:TESTDB:/home/oracle/JAVA]/u01/app/12c/jdk/bin/java -classpath
/u01/app/12c/jdbc/lib/ojdbc8.jar:. AcTest

SQL> @sess

      SID      SERIAL# USERNAME          MACHINE          MODULE
STATUS
-----
      1466      43678 ACTEST              GOODUSDATA      JDBC Thin Client
INACTIVE
      1713      34451 ACTEST              WORKGROUP\G1800007  Orange for ORACLE DBA
INACTIVE

SQL> select * from ACTEST.info;

START_DATE          ID ITEM
-----
20/01/19 18:47:39      1 Connected

```

```
SQL> alter system kill session '1466,43678';
```

```
System altered.
```

```
SQL> @sess
```

SID	SERIAL#	USERNAME	MACHINE	MODULE
1466	43678	ACTEST	GOODUSDATA	
1713	34451	ACTEST	WORKGROUP\G1800007	Orange for ORACLE DBA

```
SQL> select * from actest.info;
```

START_DATE	ID	ITEM
20/01/19 18:47:39	1	Connected

```
SQL> select * from actest.info;
```

START_DATE	ID	ITEM
20/01/19 18:47:39	1	Connected
20/01/19 18:48:39	2	Disconnected

4.3. NOREPLAY Option

- Application Continuity 가 구성되어 있고 DBA 가 alter system kill session 또는 alter system disconnect session 명령을 수행하면 Application Continuity 는 Default 로 Recover 를 수행합니다. 그러나 Session 을 Replay 하지 않고 종료하기 위해서 Noreplay 옵션을 사용 할 수 있습니다.
- JDBC Thine Client 수행

```
[GOODUSDATA:TESTDB:/home/oracle/JAVA]!762  
/u01/app/12c/jdk/bin/java -classpath /u01/app/12c/jdbc/lib/ojdbc8.jar:. AcTest  
You are Connected to Single Instance - TESTDB
```

- Session 확인 및 Kill Session(NoReplay Option)

```
SQL> @sess
```

SID	SERIAL#	USERNAME	MACHINE	MODULE
1347	61121	ACTEST	GOODUSDATA	JDBC Thin Client

STATUS

INACTIVE

```
SQL> alter system kill session '1347,61121' NOREPLAY;
```

```
System altered.
```

```
SQL> @sess
```

SID	SERIAL#	USERNAME	MACHINE	MODULE
1347	61121	ACTEST	GOODUSDATA	

STATUS

KILLED

- JDBC Thine 프로그램 확인

```
[GOODUSDATA:TESTDB:/home/oracle/JAVA]!762
```

```
/u01/app/12c/jdk/bin/java -classpath /u01/app/12c/jdbc/lib/ojdbc8.jar:. AcTest
```

```
You are Connected to Single Instance - TESTDB
```

```
Exception in thread "main" java.sql.SQLException: ORA-00045: Your session has been terminated with no replay.
```

```
at oracle.jdbc.driver.T4CTTloer11.processError(T4CTTloer11.java:494)
```

```
at oracle.jdbc.driver.T4CTTloer11.processError(T4CTTloer11.java:446)
```

```
at oracle.jdbc.driver.T4C8Oall.processError(T4C8Oall.java:1054)
```

```
at oracle.jdbc.driver.T4CTTIfun.receive(T4CTTIfun.java:623)
```

```
at oracle.jdbc.driver.T4CTTIfun.doRPC(T4CTTIfun.java:252)
```

```
at oracle.jdbc.driver.T4C8Oall.doOALL(T4C8Oall.java:612)
```

```
at oracle.jdbc.driver.T4CPreparedStatement.doOall8(T4CPreparedStatement.java:226)
```

```
at oracle.jdbc.driver.T4CPreparedStatement.doOall8(T4CPreparedStatement.java:59)
```

```
at
```

```
oracle.jdbc.driver.T4CPreparedStatement.executeForDescribe(T4CPreparedStatement.java:747)
```



```
at oracle.jdbc.driver.OracleStatement.executeMaybeDescribe(OracleStatement.java:904)
at oracle.jdbc.driver.OracleStatement.doExecuteWithTimeout(OracleStatement.java:1082)
at
oracle.jdbc.driver.OraclePreparedStatement.executeInternal(OraclePreparedStatement.java:3780)
at oracle.jdbc.driver.T4CPreparedStatement.executeInternal(T4CPreparedStatement.java:1343)
at
oracle.jdbc.driver.OraclePreparedStatement.executeQuery(OraclePreparedStatement.java:3822)
at
oracle.jdbc.driver.OraclePreparedStatementWrapper.executeQuery(OraclePreparedStatementWrapper.java:1165)
at
oracle.jdbc.proxy.oracle$1jdbc$1replay$1driver$1TxnReplayableStatement$2oracle$1jdbc$1internal$1OraclePreparedStatement$$$Proxy.executeQuery(Unknown Source)
at AcTest.main(AcTest.java:27)
Caused by: Error : 45, Position : 26, Sql = select instance_name from v$instance, OriginalSql = select instance_name from v$instance, Error Msg = ORA-00045: Your session has been terminated with no replay.

at oracle.jdbc.driver.T4CTTloer11.processError(T4CTTloer11.java:498)
... 16 more
```

- **Replay** 를 시도하지 않고 **Session** 이 중지되는 것을 확인 할 수 있습니다.

4.4. Application Contunuity for Oracle Rac

- Service 등록 및 Start

```
[goodusdb01:goodusdb1:/home/oracle]srvctl add service -s acservice -d goodusdb -r
goodusdb1,goodusdb2
[root@goodusdb02 ~]# srvctl start service -s acservice -d goodusdb
```

- Resource 확인

```
[goodusdb01:+ASM1:/home/grid]crsctl stat res -t
-----
Name          Target  State        Server          State details
-----
Local Resources
-----
ora.ASMNET1LSNR_ASM.lsnr
      ONLINE ONLINE      goodusdb01     STABLE
      ONLINE ONLINE      goodusdb02     STABLE
ora.CRSDG.dg
      ONLINE ONLINE      goodusdb01     STABLE
      ONLINE ONLINE      goodusdb02     STABLE
ora.DATA.dg
      ONLINE ONLINE      goodusdb01     STABLE
      ONLINE ONLINE      goodusdb02     STABLE
ora.LISTENER.lsnr
      ONLINE ONLINE      goodusdb01     STABLE
      ONLINE ONLINE      goodusdb02     STABLE
ora.MGMT.GHCHKPT.advm
      OFFLINE OFFLINE    goodusdb01     STABLE
      OFFLINE OFFLINE    goodusdb02     STABLE
ora.MGMT.dg
      ONLINE ONLINE      goodusdb01     STABLE
      ONLINE ONLINE      goodusdb02     STABLE
ora.RECO.dg
      ONLINE ONLINE      goodusdb01     STABLE
      ONLINE ONLINE      goodusdb02     STABLE
ora.TEST.lsnr
```

	ONLINE	ONLINE	goodusdb01	STABLE
	ONLINE	ONLINE	goodusdb02	STABLE
ora.chad				
	OFFLINE	OFFLINE	goodusdb01	STABLE
	OFFLINE	OFFLINE	goodusdb02	STABLE
ora.helper				
	OFFLINE	OFFLINE	goodusdb01	IDLE,STABLE
	OFFLINE	OFFLINE	goodusdb02	IDLE,STABLE
ora.mgmt.ghchkpt.acfs				
	OFFLINE	OFFLINE	goodusdb01	STABLE
	OFFLINE	OFFLINE	goodusdb02	STABLE
ora.net1.network				
	ONLINE	ONLINE	goodusdb01	STABLE
	ONLINE	ONLINE	goodusdb02	STABLE
ora.ons				
	ONLINE	ONLINE	goodusdb01	STABLE
	ONLINE	ONLINE	goodusdb02	STABLE
ora.proxy_advm				
	ONLINE	ONLINE	goodusdb01	STABLE
	ONLINE	ONLINE	goodusdb02	STABLE

Cluster Resources				

ora.LISTENER_SCAN1.lsnr				
1	ONLINE	ONLINE	goodusdb01	STABLE
ora.MGMTLSNR				
1	ONLINE	ONLINE	goodusdb01	169.254.18.238 10.10.10.181,STABLE
ora.asm				
1	ONLINE	ONLINE	goodusdb01	Started,STABLE
2	ONLINE	ONLINE	goodusdb02	Started,STABLE
3	OFFLINE	OFFLINE		STABLE
ora.cvu				
1	ONLINE	ONLINE	goodusdb01	STABLE
ora.goodusdb.acservice.svc				
1	ONLINE	ONLINE	goodusdb01	STABLE
2	ONLINE	ONLINE	goodusdb02	STABLE

```

ora.goodusdb.db
  1      ONLINE  ONLINE    goodusdb01      Open,HOME=/oracle/db
                                     /18.0.0.0,STABLE
  2      ONLINE  ONLINE    goodusdb02      Open,HOME=/oracle/db
                                     /18.0.0.0,STABLE
ora.goodusdb01.vip
  1      ONLINE  ONLINE    goodusdb01      STABLE
ora.goodusdb02.vip
  1      ONLINE  ONLINE    goodusdb02      STABLE
ora.mgmtdb
  1      ONLINE  ONLINE    goodusdb01      Open,STABLE
ora.qosmsserver
  1      OFFLINE OFFLINE
                                     STABLE
ora.rhpserver
  1      OFFLINE OFFLINE
                                     STABLE
ora.scan1.vip
  1      ONLINE  ONLINE    goodusdb01      STABLE
-----

```

- Enable the service for Application Continuity

```

[goodusdb01:goodusdb1:/home/oracle]srvctl modify service -d goodusdb -s acservice -failovertyp
TRANSACTION -replay_init_time 300 -failoverretry 30 -failoverdelay 3 -notification TRUE -
commit_outcome TRUE

```

- JDBC 연동 TEST

```

import java.sql.*;
import oracle.jdbc.*;
public class AcTest
{
public static void main(String[] args) throws SQLException,java.lang.InterruptedExceptio
n
{
oracle.jdbc.replay.OracleDataSource  AcDatasource =
oracle.jdbc.replay.OracleDataSourceFactory.getOracleDataSource();
AcDatasource.setURL("jdbc:oracle:thin:@ 91.91.91.181:1521/acservice");
AcDatasource.setUser("ACTEST");
AcDatasource.setPassword("welcome1");
}
}

```

```
Connection conn = AcDatasource.getConnection();

conn.setAutoCommit(false);

PreparedStatement stmt = conn.prepareStatement("select instance_name from v$instance");
ResultSet rset = stmt.executeQuery();
while (rset.next())
{
System.out.println("You are Connected to RAC Instance - "+ rset.getString(1));
}

Thread.currentThread().sleep(60000);

((oracle.jdbc.replay.ReplayableConnection)conn).beginRequest();

PreparedStatement stmt1 = conn.prepareStatement("select instance_name from v$instance");
ResultSet rset1 = stmt1.executeQuery();
while (rset1.next())
{
System.out.println("After Replay Connected to RAC Instance - "+rset1.getString(1));
}

rset.close();
stmt.close();
rset1.close();

stmt1.close();

conn.close();

((oracle.jdbc.replay.ReplayableConnection)conn).endRequest();
}
}
```

- Souece File Compile

```
[goodusdb01:goodusdb1:/home/oracle/JAVA]$ORACLE_HOME/jdk/bin/javac -classpath /oracle/db/18.0.0.0/jdbc/lib/ojdbc8.jar /home/oracle/JAVA/AcTest.java
```

```
[goodusdb01:goodusdb1:/home/oracle/JAVA]ls -lrt
```

```
합계 12
```

```
-rw-r--r--. 1 oracle dba 1268 12 월 9 10:07 AcTest_backup.java
```

```
-rw-r--r--. 1 oracle dba 1200 12 월 9 10:22 AcTest.java
```

```
-rw-r--r--. 1 oracle dba 2025 12 월 9 11:47 AcTest.class
```

```
[goodusdb01:goodusdb1:/home/oracle/JAVA]$ORACLE_HOME/jdk/bin/java -classpath /oracle/db/18.0.0.0/jdbc/lib/ojdbc8.jar:. AcTest
```

```
You are Connected to RAC Instance - goodusdb1
```

```
After Replay Connected to RAC Instance - goodusdb1
```

➔ 중간에 다른 Session 으로 접속하여 Session 을 Kill 해도 Replay 후 작업을 수행 합니다.

```
SQL> @sess
```

SID	SERIAL#	USERNAME	MACHINE	STATUS
184	4482	ACTEST	goodusdb01	JDBC

Thin Client			INACTIVE	

```
SQL> alter system kill session '184,4482';
```

```
System altered.
```

```
SQL> @sess
```

SID	SERIAL#	USERNAME	MACHINE	STATUS
184	4482	ACTEST	goodusdb01	KILLED

4.5. NoReplay Option

- Application Continuity 가 구성되어 있고 DBA 가 alter system kill session 또는 alter system disconnect session 명령을 수행하면 Application Continuity 는 Default 로 Recover 를 수행합니다. 그러나 Session 을 Replay 하지 않고 종료하기 위해서 Noreplay 옵션을 사용 할 수 있습니다.
- JDBC Thine Client 로 Database 에 Call

```
[goodusdb01:goodusdb1:/home/oracle/JAVA]$ORACLE_HOME/jdk/bin/java -classpath /oracle/db/18.0.0/jdbc/lib/ojdbc8.jar:. AcTest
You are Connected to RAC Instance - goodusdb1
```

- Session 확인 및 Kill session(Noreplay Option)

```
SQL> @sess

      SID      SERIAL# USERNAME                MACHINE
MODULE                                     STATUS
-----
-----
      26        61415 ACTEST                    goodusdb01
Thin Client                               INACTIVE                                JDBC

SQL> alter system kill session '26,61415' NOREPLAY;

System altered.

SQL> @sess

      SID      SERIAL# USERNAME                MACHINE
MODULE                                     STATUS
-----
-----
      26        61415 ACTEST                    goodusdb01
KILLED
```

- JDBC Thine 프로그램 확인

```
[goodusdb01:goodusdb1:/home/oracle/JAVA]$ORACLE_HOME/jdk/bin/java -classpath /oracle/db/18.0.0/jdbc/lib/ojdbc8.jar:. AcTest
```

You are Connected to RAC Instance - goodusdb1

^C[goodusdb01:goodusdb1:/home/oracle/JAVA]

[goodusdb01:goodusdb1:/home/oracle/JAVA]\$ORACLE_HOME/jdk/bin/java -classpath /oracle/db/18.0.0/jdbc/lib/ojdbc8.jar:. AcTest

You are Connected to RAC Instance - goodusdb1

Exception in thread "main" java.sql.SQLException: ORA-00045: 세션이 재생 없이 종료되었습니다.

at oracle.jdbc.driver.T4CTTloer11.processError(T4CTTloer11.java:494)

at oracle.jdbc.driver.T4CTTloer11.processError(T4CTTloer11.java:441)

at oracle.jdbc.driver.T4CTTIfun.receive(T4CTTIfun.java:541)

at oracle.jdbc.driver.T4CTTIfun.doRPC(T4CTTIfun.java:255)

at oracle.jdbc.driver.T4C8Oall.doOALL(T4C8Oall.java:610)

at oracle.jdbc.driver.T4CPreparedStatement.doOall8(T4CPreparedStatement.java:253)

at oracle.jdbc.driver.T4CPreparedStatement.doOall8(T4CPreparedStatement.java:86)

at

oracle.jdbc.driver.T4CPreparedStatement.executeForDescribe(T4CPreparedStatement.java:765)

at oracle.jdbc.driver.OracleStatement.executeMaybeDescribe(OracleStatement.java:921)

at oracle.jdbc.driver.OracleStatement.doExecuteWithTimeout(OracleStatement.java:1099)

at oracle.jdbc.driver.OraclePreparedStatement.executeInternal(OraclePreparedStatement.java:3640)

at oracle.jdbc.driver.T4CPreparedStatement.executeInternal(T4CPreparedStatement.java:1384)

at

oracle.jdbc.driver.OraclePreparedStatement.executeQuery(OraclePreparedStatement.java:3687)

at

oracle.jdbc.driver.OraclePreparedStatementWrapper.executeQuery(OraclePreparedStatementWrapper.java:1165)

at

oracle.jdbc.proxy.oracle\$1jdbc\$1replay\$1driver\$1TxnReplayableStatement\$2oracle\$1jdbc\$1internal\$1OraclePreparedStatement\$\$\$Proxy.executeQuery(Unknown Source)

at AcTest.main(AcTest.java:27)

- Replay 를 시도하지 않고 Session 이 중지되는 것을 확인 할 수 있습니다.

5. Potential Side Effects of Application Continuity

- Session 이 Rebuild 되면서 모든 Status 가 Rebuild 되기 때문에 Side Effect 가 발생 할 수 있습니다.
 - Autonomous transactions
 - DBMS_ALERT Calls
 - DBMS_FILE_TRANSFER Calls
 - DBMS_PIPE and RPC Calls

-
- UTL_FILE Calls
 - UTL_HTTP Calls
 - UTL_MAIL Calls
 - UTL_SMTP Calls
 - UTL_TCP Calls
 - UTL_URL Calls

6. Application Continuity 제약사항

- JDBC Thin Connection 에만 적용됩니다 (JDBC OCI 는 지원되지 않음).
- For applications using JDBC, there is no support for oracle.sql deprecated concrete classes: BLOB, CLOB, BFILE, OPAQUE, ARRAY, STRUCT, or ORADATA. (See My Oracle Support Note 1364193.1)
- Oracle XA 를 사용하여 개발 된 애플리케이션에는 Replay 가 지원되지 않습니다.
- ALTER SYSTEM 또는 ALTER DATABASE 문에 대해서는 Replay 가 비활성화 됩니다 .
- statement-level failure 인 경우 병렬 쿼리 호출 실패에 대해서는 재생이 적용되지 않습니다.
Ex) replay 는 ORA-12805 error ("parallel query server died unexpectedly") 에러에 대해 인스턴스 또는 노드 장애 또는 메모리 문제 중에 발생한 호출 실패에 대한 Replay 가 수행되지 않습니다.

7. References

- https://docs.oracle.com/database/121/ADFNS/adfn_app_continuity.htm#ADFNS1058
- https://docs.oracle.com/database/121/ARPLS/d_serv.htm#ARPLS092
- **How To Test Application Continuity Using A Standalone Java Program (문서 ID 1602233.1)**