An Oracle White Paper
September 2010

Application Failover with Oracle Database 11g

ORACLE®

Summary

This paper is a brief overview of high availability mechanisms built into the Oracle database server and its clients (i.e., drivers, adapters) to support application failover, during planned as well as unplanned database outages. The primary goal of this paper is to clarify Oracle's HA features FAN, FCF and TAF and make recommendations on when to use which. In a follow-up paper, we will discuss in greater details how to use these mechanisms.

Oracle Database HA Configurations

The Oracle Database furnishes the following High Availability configurations:

- Single Instance HA

- Cold Failover Cluster

- RAC, RAC One, RAC with Vendor Clusterware

- Data Guard Physical Standby (single Instance or RAC with/without Broker)

- Data Guard Logical Standby (Single Instance or RAC with/without Broker)

Detailed coverage of each of these configurations can be found in the Oracle Database 11g Release 2 High Availability Overview document @ http://download.oracle.com/docs/cd/E14072_01/server.112/e10804.pdf

The Why and What of Application Failover

When a database outage occurs, two problems (the Evil Twins) confront applications: errors and hangs. Applications encounter errors, because the work they were doing (queries, transactions) is interrupted. Even worse, those errors may take some time to arrive. When the database outage occurs, the client application may have to wait for network timeouts (which may be minutes) before being informed of the outage. This can cause the application to hang for some time, leading to user frustration.

Oracle's HA features address these Twin Evils by helping to speed application response

to failure, and by helping to mask the error from the end user in some situations

Fast Application Notification (FAN)

FAN addresses one of the Evil Twins: hangs. FAN is an Oracle High Availability mechanism that emits events when database conditions change i.e., a managed service, instance or site[1] goes up or down. The events are propagated by either Oracle Notification System (ONS) to Java subscribers, or Streams AQ to OCI (C, C++, PHP, Python) and .NET subscribers. The main benefits of FAN when compared to TCP timeouts are: fast detection of condition change and fast notification.
FAN is available through the following Oracle components: CMAN session pools; Oracle Call Interface (OCI) and a number of drivers or adapters that use OCI libraries (including OCCI, PHP, Python); Universal Connection Pool for Java; JDBC SimpleFAN API; and ODP.NET connection pools.
How to use Fan events?

a) Non-programmatically through Oracle database integrated clients (using Oracle Restart): Oracle JDBC, Universal Connection Pool for Java, Oracle Call Interface, and ODP.NET. These clients can be configured to enable FAN and automatically connect to a new primary database upon failover using Fast Connection Failover (FCF), see later.

b) Programmatically: 3rd party drivers, containers or frameworks may use Java FAN API (SimpleFAN) or OCI Callbacks, to handle FAN events themselves.

c) FAN callout scripts can be configured on the database tier and allow server-side actions when conditions change.

In an upcoming paper, we will furnish a more detailed and complete coverage of how to use FAN.

Transparent Application Failover (TAF) Overview

Transparent Application Failover (TAF) helps to address the other Evil Twin: errors. TAF is an OCI feature providing connection recovery capabilities: connection failover, session state restoration, query failover, and graceful session migration for planned downtime[2].
TAF operates at session or connection level and is available to database clients that use the OCI driver including: OCI, OCCI, Pre-compilers (Pro*), ODP.Net, JDBC-OCI (not JDBC thin), PHP OCI8, Ruby OCI8, Python cx-oracle, etc.
TAF is particularly useful for read-only and read-mostly applications. When failure occurs amid a query or transaction and the client has subscribed to FAN events, TAF will re-

---

[1] After a failover, the Oracle Data Guard Broker (broker) publishes FAN events.

[2] See the TAF Best Practices white paper @ … for a more detailed explanation of TAF features.

execute the query and re-position the cursor. As a result, the application may continue fetching after a failure occurs. If failure occurs during a transaction, the database rolls back the transaction, TAF notifies the client to clean up application state (by issuing ROLLBACK) before resuming normal operations on a new connection.

TAF may be used with or without FAN.
a) In TAF-only environments (i.e., when not subscribing to FAN events), upon condition change (i.e., node down), and upon expiration of TCP timeout, TAF executes the recovery and failover process; the client may experience longer delay (varies with systems) because, unlike FAN, application threads may remain blocked until the TCP timeout expires.

b) When combined with FAN, in RAC and Data Guard environments, delays due to TCP timeouts are eliminated.

c) TAF callback functionality[3] (covered in details in upcoming paper), allows applications to extend the TAF recovery mechanism.

Fast Connection Failover (FCF)

FCF designates the set of actions that *integrated Oracle clients* (UCP, OCI Session pool, etc) take to process FAN events. The key features of FCF are:
a) Rapid database service/instance/node failure detection then abort and removal of invalid connections from the pool

   Unplanned outages -- Dead connections are rapidly detected: the borrowed and in-use connections are aborted and removed from the pool; idle connections are cleaned up as well.
   Planned Outages (graceful shutdown) -- Borrowed or in-use connections are not interrupted however, at the completion of the database operation the connections are marked for removal and returned to the pool. When all connections are checked-in, back to the pool, the database can shutdown gracefully.
b) Recognition of new nodes that join an Oracle RAC cluster

c) Runtime distribution of connection requests to all active Oracle RAC instances

Recommendations

---

[3] http://download.oracle.com/docs/cd/E14072_01/appdev.112/e10646.pdf

The following rules of thumb or recommendations apply:

1. When using integrated Oracle clients (JDBC, OCI, .Net etc), FAN with FCF is highly recommended as it provides immunity from TCP timeouts for in-flight calls, and eagerly cleans up dead connections from connection pools to minimize application exposure to the failure.

2. For - read-only and read-mostly applications, TAF (in conjunction with FAN) is the recommended choice; it provides query failover (i.e. it allows active queries to continue) without disrupting the application.

3. Java containers, drivers, frameworks, or applications may use FAN API to directly manage FAN events themselves. OCI-based containers, drivers, frameworks or applications (C, C++, OCCI, PreComps, PHP, Ruby, Python, Perl) may use OCI callbacks to directly manage FAN events themselves.

ORACLE®

Oracle is committed to developing practices and products that help protect the environment