77
DBMS_DG

The `DBMS_DG` package allows applications to notify the primary database or the fast-start failover target database in an Oracle Data Guard broker environment to initiate a fast-start failover when the application encounters a condition that warrants a failover.

> **Note:**
>
> A multitenant container database is the only supported architecture in Oracle Database 21c and later releases. While the documentation is being revised, legacy terminology may persist. In most cases, "database" and "non-CDB" refer to a CDB or PDB, depending on context. In some contexts, such as upgrades, "non-CDB" refers to a non-CDB from a previous release.

This chapter contains the following topics:

> **See Also:**
>
> *Oracle Data Guard Broker*

Using DBMS_DG

There are conditions detectable by applications running outside of the Oracle database that may warrant the Oracle Data Guard broker to perform a fast-start failover. Because the range of possible conditions is virtually unlimited, it is left to the applications to determine which conditions warrant a fast-start failover.

When such conditions occur, the application calls the `DBMS_DG.INITIATE_FS_FAILOVER` procedure to alert either the primary or fast-start failover target standby database that the application wants a fast-start failover to occur immediately. The database on which the procedure was called then notifies the observer, which immediately initiates a fast-start failover as long as the standby database is in a valid fast-start failover state ("observed" and either "synchronized" or "within lag") to accept a failover.If the configuration is not in a valid fast-start failover state, the `INITIATE_FS_FAILOVER` subprogram returns an ORA error message (it will not signal an exception) to inform the calling application that a fast-start failover could not be performed.

> **Note:**
>
> If you are working in a multitenant container database (CDB), then functions within DBMS_DG are only executed at the root level. Ensure you are connected at the root level, not at the individual pluggable database (PDB) level.

DBMS_DG Security Model

The DBMS_DG package runs with invoker's rights and requires the SYSDBA privilege.

Summary of the DBMS_DG Subprogram

The DBMS_DG package contains one subprogram, the INITIATE_FS_FAILOVER procedure.

Table 77-1 DBMS_DG Package Subprogram

Subprogram	Description
INITIATE_FS_FAILOVER Procedure	Enables an application to notify either the primary or fast-start failover target standby database that a fast-start failover is necessary when the application encounters conditions that warrant a failover. This procedure can only be called while connected to a primary database or a fast-start failover standby database.

INITIATE_FS_FAILOVER Procedure

Use this procedure to specify a condition string that, when encountered by an application, allows the application to request that a fast-start failover be invoked.

Syntax

```
DBMS_DG.INITIATE_FS_FAILOVER (
     condstr          IN VARCHAR2)
RETURN BINARY_INTEGER;
```

Parameters

Table 77-2 INITIATE_FS_FAILOVER Procedure Parameters

Parameter	Description
condstr	Specifies the condition string for which a fast-start failover should be requested. If no condition string argument is supplied, the default string of "Application Failover Requested" will be logged in the broker log file and in the database alert log of the database on which the procedure was called.

Usage Notes

- This procedure returns a binary integer.
- Query the V$FS_FAILOVER_STATS view to see the time of the last fast-start failover and the reason it was performed.

- This procedure can only be called while connected to a primary database or a fast-start failover standby database.

Errors

Table 77-3 INITIATE_FS_FAILOVER Procedure Errors

Error	Description
`ORA-00000: normal, successful completion`	The request to initiate a fast-start failover has been posted to the observer.
`ORA-16646: fast-start failover is disabled`	Either a broker configuration does not exist or fast-start failover has not been enabled.
`ORA-16666: unable to initiate fast-start failover on a bystander standby database`	`DBMS_DG.INITIATE_FS_FAILOVER` was invoked on a bystander standby database. That is, it was not invoked on the primary or on the fast-start failover target standby database.
`ORA-16817: unsynchronized fast-start failover configuration`	`DBMS_DG.INITIATE_FS_FAILOVER` was invoked in a maximum available fast-start failover configuration when the configuration was not synchronized.
`ORA-16819: fast-start failover observer not started`	`DBMS_DG.INITIATE_FS_FAILOVER` was invoked but an observer had not yet been started.
`ORA-16820: fast-start failover observer is no longer observing this database`	`DBMS_DG.INITIATE_FS_FAILOVER` was invoked but the configuration detects that the observer may not be running.
`ORA-16829: lagging fast-start failover configuration`	`DBMS_DG.INITIATE_FS_FAILOVER` was invoked in a maximum performance fast-start failover configuration when the configuration was not in the user-specified redo lag limit.

Example

In this example, the program attempts to initiate a fast-start failover when fast-start failover is disabled. To use this example, connect as user `SYS` with `SYDDBA` privileges.

```
set serveroutput on

declare
status integer;

begin
status := dbms_dg.initiate_fs_failover(''Failover Requested'');

dbms_output.put_line(''Fast-Start Failover is disabled: Expected status = ORA-16646'');
dbms_output.put_line(''                 Actual Status = ORA-'' || status);

end;
/
exit;
```