**https://docs.oracle.com/cd/B19306\_01/B14251\_01/adfns\_dcn.htm**

**What Is Database Change Notification?**

receive notifications in response to DML or DDL changes 。The notifications are published by the database when the DML or DDL transaction commits.

A notification handler can be either a server side PL/SQL procedure or a client side C callback.

When the database issues change notification, it can contain some or all of the following information:

* Names of the modified objects. For example, the notification can specify that the hr.employees table was changed.
* The type of change. For example, the message specifies whether the change was caused by an INSERT, UPDATE, DELETE, ALTER TABLE, or DROP TABLE.
* The ROWIDs of the changed rows and the type of DML that changed them.
* Global events such as STARTUP and SHUTDOWN (consistent only). In a Real Applications Cluster, the database delivers a notification when the first instance on the database starts or the last instance shuts down.

* The notification includes information on the names of the objects changed, the Transaction-Id of the transaction that made the change and the TYPE of operation (INSERT, UPDATE or DELETE).

The notification contains only metadata about the changed rows or objects rather than the changed data itself.

只通知变化实体和行，不通知列变化

Example of Mid-Tier Caching

4.Oracle Database adds a message that describes the change to an internal queue.

5. A JOBQ background process is notified of a new change notification message.

6. The JOBQ process executes the stored procedure specified by the client application.

7. Inside the server-side PL/SQL procedure, the developer can implement logic to notify the mid-tier client application of the changes to the registered objects.

## Registering Queries for Database Change Notification

In order to create a registration for change notification, the user is required to have the CHANGE NOTIFICATION system privilege.

Once created, the registration survives until explicitly unregistered by the client application or timed-out or implicitly removed by the database for some other reason (such as loss of privileges).

1.Create the notification recipient

. The recipient can be one of the following:

* PL/SQL stored procedure, as described in ["Creating a PL/SQL Stored Procedure as the Change Notification Recipient"](https://docs.oracle.com/cd/B19306_01/B14251_01/adfns_dcn.htm#BGBDBAIJ)
* OCI callback function,

Change Notification allows the application to register most query types including queries executed as part of stored procedures and REF cursors.

In addition, the following types of queries are not supported for registration.

* Queries on fixed tables or fixed views.
* Queries with dblinks inside them
* Queries over materialized views

### Registration Properties

Oracle Database supports the following options for an object registration:

1. Purge On Notify option: Unregistering after the first change notification.
2. Timeout option: Specification of a registration expiration after a time interval.
3. ROWIDs option: ROWIDs of changed rows are part of the notification ROWID option.
4. …
5. Operations filter: Ability to be notified of PARTICULAR operations (for example notifications only for INSERT AND UPDATE).
6. Transaction Lag: Specification of a count between successive notifications.

then the notification might be rolled up into a FULL-TABLE-NOTIFICATION (a special flag in the notification descriptor is reserved for this purpose). When such a notification is received, the application must conservatively assume that the entire table (that is, all rows) may have been modified.

When a table is dropped, a DROP NOTIFICATION is published. Any registrations on the dropped table will implicitly remove interest from that object (since it does not exist anymore). If those registrations have interest in other objects as well, then the registrations will continue to exist and DML transactions on those other objects will continue to result in notifications on commit.

一次可以注册多个实体的通知？

Even if the dropped table is the only object of interest for a particular registration, we still preserve the registration. The user that created that registration can use the registration to add more objects/queries subsequently.

If an object was dropped, registrations on the object will lose interest on the object forever.

even if a different object was created with a matching name and in the same schema, any changes to this newly created object (with the matching schema/name) will not result in notifications for those registrations