



1. TRANSPARENCIA DE DISTRIBUCIÓN PARA INSTRUCCIONES DML

1. TRANSPARENCIA DE DISTRIBUCIÓN PARA INSTRUCCIONES DML.....	1
1.1. Objetivo.....	2
1.2. Actividades previas.....	2
1.3. Implementación de transparencia dml y transaccionalidad.....	2
1.3.1. Privilegios para crear triggers.....	2
1.3.2. Definición de tablas temporales para manejo de columnas BLOB.....	2
1.3.3. Definición del trigger Instead Of para la vista pais.....	2
1.3.4. Definición del trigger Instead Of para la vista oficina.....	3
1.3.5. Definición del trigger instead of para la vista empleado.....	3
1.3.6. Ejecución de triggers.....	3
Ejemplo.....	3
1.4. Validación de resultados.....	4
1.5. Contenido del reporte.....	5

1.1. Objetivo

Comprender y poner en práctica estrategias para implementar los niveles de transparencia de distribución y transaccionalidad en una base de datos distribuida para operaciones DML.

1.2. Actividades previas.

Revisar el documento [ejercicio-practico-08-previo.pdf](#)

1.3. Implementación de transparencia dml y transaccionalidad

Considerar nuevamente el esquema de fragmentación y la solución de fragmentación para la instrucción select realizadas en ejercicios anteriores. En esta ocasión se realizará la transparencia de operaciones sólo para los siguientes escenarios:

- Transparencia para operaciones **insert** y **delete** solo para las tablas **país**, **oficina** y **empleado**. Para operaciones **update** el trigger deberá lanzar una excepción indicando que la operación update aún no está soportada.
- Transparencia para operaciones **insert**, **delete** y **update** únicamente para la tabla **pais**.

1.3.1. Privilegios para crear triggers

- Crear Scripts SQL **s-01-<iniciales>-privilegios-usuarios.sql**
- En esta ocasión se requieren permisos en ambas PDBs para que el usuario correspondiente pueda crear triggers.
- El script se deberá conectar a cada PDB y como usuario **SYS** otorgar el privilegio correspondiente.

1.3.2. Definición de tablas temporales para manejo de columnas BLOB

- Crear Scripts SQL **s-02-<iniciales>-tablas-temporales.sql**
- El script deberá definir 2 tablas temporales: **t_empleado_insert** y **t_empleado_update** que serán empleadas por el trigger que implementará transparencia DML.
- El script deberá conectarse únicamente a las PDBs que requieran hacer accesos remotos para obtener un objeto BLOB.

1.3.3. Definición del trigger Instead Of para la vista pais

- Crear Scripts SQL **s-03-<iniciales>-pais-[n<{1|2}>]-trigger.sql**
- Notar que el término **[n<{1|2}>]** se puede omitir ya que el código de este trigger es idéntico en ambos nodos. Solo se requiere un script por integrante. El nombre del archivo será **s-03-<iniciales>-pais-trigger.sql**

- El trigger deberá lanzar el código de error -20001 si el registro no cumple con los predicados de fragmentación.
- **C1. Incluir en el reporte** únicamente el código donde se implementa la instrucción `update` del “Integrante 1”. No olvidar dar formato al código.

1.3.4. Definición del trigger Instead Of para la vista oficina

- Crear Scripts SQL `s-03-<iniciales>-oficina-n<{1|2}>-trigger.sql`
- Observar que se requiere un archivo para cada PDB ya que el código del trigger es diferente al tratarse de una fragmentación horizontal derivada. El trigger deberá verificar en el fragmento local y posteriormente en el modo remoto.
- El trigger deberá lanzar el código de error -20001 si el registro no cumple con los predicados de fragmentación.
- El trigger deberá lanzar el código de error -20002 si se intenta realizar una operación `update`.
- **C2. Incluir en el reporte** el código donde se implementa la instrucción `insert`.

1.3.5. Definición del trigger instead of para la vista empleado.

- Crear scripts `s-03-<iniciales>-empleado-n<{1|2}>-trigger.sql` Observar que en este caso si se requiere un trigger diferente por cada nodo debido al manejo del dato `blob` y al esquema horizontal.
- **C3. Incluir en el reporte** únicamente el código donde se implementa la instrucción `insert`.
- El trigger deberá lanzar el código de error -20001 si el registro no cumple con los predicados de fragmentación.
- El trigger deberá lanzar el código de error -20002 si se intenta realizar una operación `update`.

1.3.6. Ejecución de triggers.

Crear Scripts SQL `s-04-<iniciales>-main-triggers.sql` El script se deberá conectar a cada PDB y ejecutar la compilación y creación de los triggers.

Ejemplo

```
--@Autor:          Jorge A. Rodriguez C
--@Fecha creación: dd/mm/yyyy
--@Descripción:    Creación de triggers
Prompt creando triggers en S1
connect consultora_bdd/consultora_bdd@jrcbdd_s1

Prompt creando trigger para pais
@s-03-jrc-pais-trigger.sql
```

```
show errors
```

```
Prompt creando trigger para oficina
```

```
@s-03-jrc-n1-oficina-trigger.sql
```

```
show errors
```

```
Prompt creando trigger para empleado
```

```
@s-03-jrc-n1-empleado-trigger.sql
```

```
show errors
```

```
Prompt creando triggers en S2
```

```
connect consultora_bdd/consultora_bdd@jrcbd_s2
```

```
Prompt creando trigger para pais
```

```
@s-03-jrc-pais-trigger.sql
```

```
show errors
```

```
Prompt creando trigger para oficina
```

```
@s-03-jrc-n2-oficina-trigger.sql
```

```
show errors
```

```
Prompt creando trigger para empleado
```

```
@s-03-jrc-n2-empleado-trigger.sql
```

```
show errors
```

```
Prompt Listo!
```

```
exit
```

Observar que se usa la instrucción `show errors` para mostrar posibles errores de compilación.

1.4. Validación de resultados

- En la carpeta compartida del ejercicio obtener todos los scripts sql/plb.
- Editar el script `s-06-validador-main.sql` con los valores correspondientes en caso de ser necesario.
- En una nueva terminal cambiarse al directorio donde se encuentran los scripts, y ejecutar el script editado (no se requiere emplear al usuario Oracle).

```
sqlplus /nolog
```

```
start s-06-validador-main.sql
```

En caso de existir errores, revisar y leer cuidadosamente los mensajes de error, corregir y reintentar. **C4. Incluir en el reporte** la captura de pantalla con la salida del validador.

1.5. Contenido del reporte.

- Elementos comunes. Para mayores detalles revisar el documento de instrucciones generales para realizar ejercicios prácticos
- C1. Extracto del trigger para **pais** que implementa la sentencia **update**
- C2. Extracto del trigger para **oficina** que implementa la sentencia **insert**
- C3. Extracto del trigger para **empleado** que implementa la sentencia **insert**
- C4. Salida del Validador.