Practice - Fast Ingest

Practice Target

Demonstrate Fast Ingest feature in Oracle database 23ai.

Practice Overview

In this practice you will demonstrate using Fast Ingest feature in Oracle database 23ai.

Demonstrating Fast Ingest Feature

1. Open SQL Developer and connect to the PDB as **sys** user

2. Show the parameter values for the MemOptimize write area.

   The default is zero which means the database will determine the size based on memory available. Setting this parameter requires restarting the database instance.

```
show parameter MEMOPTIMIZE_WRITE_AREA_SIZE
```

3. Open Putty and connect to the PDB as HR. Set up a table for memoptimize for write.

```
sql hr/oracle@//localhost:1521/freepdb1

CREATE SEQUENCE hr.clickstream_seq
  START WITH 1
  INCREMENT BY 1
  NOCACHE;

CREATE TABLE hr.user_clickstream (
  click_id   NUMBER
             DEFAULT hr.clickstream_seq.NEXTVAL
PRIMARY KEY,
  user_id    NUMBER       NOT NULL,   page_url
VARCHAR2(200),
  click_time TIMESTAMP(3)  DEFAULT SYSTIMESTAMP
)
MEMOPTIMIZE FOR WRITE;

-- verify:
SELECT table_name,  memoptimize_write
  FROM user_tables
 WHERE table_name = 'USER_CLICKSTREAM';
```

4. Simulate a high-velocity burst by inserting 1000 rows.

```
BEGIN
  FOR i IN 1..1000 LOOP
    INSERT /*+ MemOptimize_Write */
    INTO hr.user_clickstream(user_id, page_url)
    VALUES (MOD(i,100)+1, '/catalog?p='||MOD(i,50));
  END LOOP;
  COMMIT;
END;
/
```

5. Check how many rows are there in the table.

   You might see no row in the table as the inserted rows are still buffered in the memory.

```
select count(*) from hr.user_clickstream;
```

6. Check the memoptimize high water mark value.

   The write HWM must be incremented for the current session.

```
SELECT
  TO_NUMBER(dbms_memoptimize.get_write_hwm_seqid()) AS write_hwm,
  TO_NUMBER(dbms_memoptimize.get_apply_hwm_seqid()) AS flushed_hwm FROM
dual;
```

7. Explicitly flush the memoptimize write area and let the data be written to disk

```
exec DBMS_MEMOPTIMIZE.WRITE_END
```

8. Verify that data has been written to the table

```
select count(*) from hr.user_clickstream;
```

9. Simulate a high-velocity burst by inserting 1000 rows without using the hint. Flush the memory afterwards.

```
ALTER SESSION SET memoptimize_writes = ON;
BEGIN
  FOR i IN 1..1000 LOOP
    INSERT
    INTO hr.user_clickstream(user_id, page_url)
    VALUES (MOD(i,100)+1, '/catalog?p='||MOD(i,50));
  END LOOP;
  COMMIT;
END;
/  exec
sys.dbms_session.sleep(1) exec
DBMS_MEMOPTIMIZE.WRITE_END
```

10. Verify that data has been written to the table

```
select count(*) from hr.user_clickstream;
```

11. In SQL Developer (SYS session), check on the memory sizes allocated for the Fast Ingest transactions.

```
SELECT  TOTAL_SIZE,
 USED_SPACE,
 FREE_SPACE,
 NUM_WRITES,
 NUM_WRITERS,
 CON_ID
FROM v$memoptimize_write_area;
```

12. In Putty session, as a cleanup drop the table and the sequence.

```
DROP TABLE hr.user_clickstream PURGE;
DROP SEQUENCE hr.clickstream_seq;
```