April 1st, 2004

ORACLE FLASHBACK – 9i To 10g

Inderpal S. Johal
Principal Consultant
Data Softech Inc.
AGENDA

- Why Flashback
- Oracle 9i Flashback Query [ Rel 1 & 2 ]
- Flashback feature in 10g Release 1
- Questions and Answers
Why Flashback

- Human Error is the biggest single cause of downtime that resulted to flashback.

- Need to quickly determine what happened and fix it to avoid downtime.

- Traditional recovery is slow and complex

- Auditing
# Oracle User Errors Recovery

<table>
<thead>
<tr>
<th>Oracle Release</th>
<th>Available options</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Oracle 8</strong></td>
<td>Point in Time Recovery</td>
</tr>
<tr>
<td><strong>Oracle 8i</strong></td>
<td>LogMiner</td>
</tr>
<tr>
<td><strong>Oracle 9i Rel 1</strong></td>
<td>Flashback Query – DBMS_Flashback</td>
</tr>
<tr>
<td><strong>Oracle 9i Rel 2</strong></td>
<td>Flashback Query</td>
</tr>
<tr>
<td><strong>Oracle 10g Rel 1</strong></td>
<td>Flashback Database</td>
</tr>
<tr>
<td></td>
<td>Flashback Drop</td>
</tr>
<tr>
<td></td>
<td>Flashback Table</td>
</tr>
<tr>
<td></td>
<td>Flashback Version query</td>
</tr>
<tr>
<td></td>
<td>Flashback Transaction Query</td>
</tr>
</tbody>
</table>
ORACLE 9i Flashback Query

Topics

UNDO Segments
Flashback Query General Feature
Flashback Query Restriction
Flashback Query Feature [Rel 1]
Flashback Query Feature [Rel 2]
UNDO Vs Rollback Segment

Rollback Segments

- RBS1
- Rbs2

Default Tablespace
Manual Management

UNDORollback Segments

Manual Management

UNDO Segments
Internally Managed
Dynamic space Mgmt.
Reserved for system managed objects

SQL> CREATE UNDO TABLESPACE undotbs1
    datafile '/u01/undotbs01.dbf'
    Size 100M autoextend on
    RETENTION GUARANTEE;
Key UNDO Init.ora parameter

- **UNDO_MANAGEMENT (MANUAL or AUTO)**
  Specifies whether or not to use AUM. Default = MANUAL

- **UNDO_TABLESPACE = \(<tablespace_name>\)**
  Specifies which undo tablespace to use.

- **UNDO_RETENTION (in seconds default = 15min)**
  Specifies how long to keep undo.

- **UNDO_SUPPRESS_ERRORS (TRUE | FALSE)**
  Used to avoid errors when “SET TRANSACTION USE ROLLBACK SEGMENT” is issued. Default = TRUE
Flashback Query [1]

Provide Consistent View of Database at a point in the past

If UNDO blocks are available

Query using

SCN Timestamp

Flashback_SCN Flashback_Time

Faster than LogMiner
Flashback Query [2]

Timestamp \(\rightarrow\) SCN

**SMON_SCN_TIME**
- Populated every 5 min
- Keep 5 days of Record

```sql
SELECT time_dp, scn_bas
FROM smon_scn_time;
```

<table>
<thead>
<tr>
<th>TIME_DP</th>
<th>SCN_BAS</th>
</tr>
</thead>
<tbody>
<tr>
<td>24-mar-04 03:15:36</td>
<td>3892362020</td>
</tr>
<tr>
<td>24-mar-04 03:20:43</td>
<td>3892362507</td>
</tr>
<tr>
<td>24-mar-04 03:25:50</td>
<td>3892362990</td>
</tr>
<tr>
<td>24-mar-04 03:30:58</td>
<td>3892363448</td>
</tr>
</tbody>
</table>
Flashback Query [3]

- **TIMESTAMP_TO_SCN**
- **SCN_TO_TIMESTAMP**

Convert the SCN to Timestamp and vice versa

SQL>`SELECT SCN_TO_TIMESTAMP(current_scn) FROM v$database;`

SCN_TO_TIMESTAMP(CURRENT_SCN)
-----------------------------------------
31-JAN-03 01.48.31.0000000000 PM
Flashback Query Restriction

Using Timestamp

Inappropriate SCN

V$Views

Database Links

PlSql Package/Procedure

When transaction is Active

Cannot UNDO for Object with Recent DDL
Flashback Query [9i Release 1]

- Utilized via the DBMS_FLASHBACK package
- Grant execute on dbms_flashback to the user or roles who want to use this features

```
SQL> GRANT EXECUTE ON DBMS_FLASHBACK TO SCOTT;
```

- Use DBMS_FLASHBACK package
  - Enable_at_time - Procedure
  - Enable_at_system_change_number - Procedure
  - Disable – Procedure
  - Get_system_change_number - Function
SQL> DELETE FROM emp;
    15 rows deleted.
SQL> COMMIT;
    Commit complete.
SQL> SELECT COUNT(*) FROM emp;
   0 rows  Now Wait for 5 Minutes
SQL> exec dbms_flashback.enable_at_time(sysdate - 5/1440);
SQL> SELECT COUNT(*) FROM emp;
   15 rows
SQL> execute DBMS_FLASHBACK.DISABLE;
SQL> SELECT COUNT(*) FROM emp;
   0 rows
DECLARE
CURSOR c IS SELECT * FROM users;
v_rec c%ROWTYPE;
BEGIN
Commit;
DBMS_FLASHBACK.DISABLE;
DBMS_FLASHBACK.Enable_AT_TIME ('22-MAR-04 10:05:08');
OPEN c;
DBMS_FLASHBACK.DISABLE;
LOOP
  FETCH c INTO v_row;
  EXIT WHEN c%NOTFOUND;
  INSERT INTO users VALUES (v_rec.user_id, v_rec.first_name, v_rec.last_name, v_rec.email);
END LOOP;
CLOSE c;
COMMIT;
END;

Flashback Query [9i Release 1]

- Cannot be used by user SYS as it cannot make calls to `dbms_flashback` package.
- No DDL or DML allowed when the session is enabled for Flashback Mode until it is not disabled.
- PL/SQL cursors opened in flashback mode are available for DML after flashback mode is disabled.
- Flashback queries against MV do not take advantage of query rewrite.
Flashback Query [9i Release 2]

SQL> SELECT * FROM DEPT
    AS OF TIMESTAMP
    to_timestamp('03-30-2004 10:00:00', 'DD-MM-YYYY hh24:mi:ss');

OR

SQL> CREATE TABLE DEPT_OLD AS
    SELECT * FROM DEPT AS OF TIMESTAMP
    to_timestamp('03-30-2004 10:00:00', 'DD-MM-YYYY hh24:mi:ss');

- No need to use any PlSql Cursor to use DML/DDL
- No Need to use the Dbms_flashback package to enable the session with SCN or past time
Tips for Using Flashback Query

- Generate current statistics on tables involved in flashback queries as FB use CBO.
- Use the AS OF SCN clause where the data is critical, so as to avoid 5-minute rounding error.
- Use Indexes while selecting small sets of data with FBQ. If you must do a FTS, use PARALLEL hint.
- Use the AS OF clause if performing single FBQ.
- Use calls to the DBMS_FLASHBACK package when you want to use the same past time for several consecutive queries.
ORACLE 10g Flashback Technology

Topics

Flashback Recovery Area
Flashback Logs
Recycle bin
Flashback Database
Flashback Drop
Flashback Table
Flashback Version Query
Flashback Transaction Query
Flashback Error Correction

- **Flashback Database**
  - Restore database to time
  - Uses flashback logs

- **Flashback Drop**
  - Restore dropped table
  - Uses recycle bin

- **Flashback Table**
  - Restore all rows in table to time
  - Uses UNDO in database

- **Flashback Transaction Query**
  - Query a committed Txn

- **Flashback Versions Query**
  - Query changes to rows over time
ORACLE 10g Flashback DATABASE

**Pre-requisite**
- Database in ARCHIVELOG mode
- Flash Recovery Area Enabled
Configuring Flashback Database

```
CONFIGURE the FLASH RECOVERY AREA

db_recovery_file_dest

SQL> STARTUP MOUNT;

SQL> ALTER DATABASE ARCHIVELOG;
SQL> ALTER DATABASE FLASHBACK ON;
SQL> SELECT FLASHBACK_ON FROM V$DATABASE;
```

1 in bytes
2 in minutes & default is 1440 minutes
Using Oracle Managed File locations will greatly simplify the administration of the database. Oracle will automatically create and delete database files as needed under specified Database and Flash Recovery areas.

- **Use Database Area for all database files**
  
  Database Area: `{ORACLE_BASE}/oradata/{DB_NAME}`  
  
  It is recommended that online redo logs and control files be written to multiple locations spread across different disks to provide greater fault tolerance.

- **Use Flash Recovery Area for all backup and recovery files**
  
  Oracle recommends that the database files and recovery files be located on physically different disks for data protection and performance.

  - Flash Recovery Area: `{ORACLE_BASE}/flash_recovery`  
  - Flash Recovery Area Size: 2048 MBytes

- **Enable Archive Log Mode**

- **Enable Backup**

  Backup Start Time: 12:00 AM

File Location Variables...
Flash Recovery Area

It is highly recommended that you use flash recovery area to automate your disk backup management.

Flash Recovery Area Location /INDY/ora10g/product/oradata/ora10/

Flash Recovery Area Size 2 GB

Flash Recovery Area Size must be set when the location is set

Used Flash Recovery Area Size (MB) 7.812

Enable flashback logging for fast database point-in-time recovery

The flash recovery area must be set to enable flashback logging. When using flashback logs, you may recover your entire database to a prior point-in-time without restoring files. Flashback is the preferred point-in-time recovery method in the recovery wizard when appropriate.

Specify how far back you wish to flash the database in the future

Flashback Retention Time 24 Hours

Current size of the flashback logs(MB) 7.812

Lowest SCN in the flashback data 1467514

Time of the lowest SCN in the flashback data Mar 23, 2004 11:05:12 AM

Apply changes to SPFILE only. Otherwise the changes will be made to both SPFILE and the running instance which requires that you restart the database to invoke static parameters.

TIP * indicates controls, if changed, must restart database to invoke.
Flashback Recovery Area Components files

**Multiplexed Copy of Control file** – If FRA destination is specified in `Control_files` parameter

**Online Redo Logs** – If member is created in FRA destination

**Control file Autoback** – RMAN can create it if configured
Configure controlfile autobackup format for device type disk clear;

** Archived log files** – Except `log_archive_dest_10`, use
`Log_archive_dest_n=‘location=use_db_recovery_file_dest’`

**Flashback logs** – Created by default when FRA is enabled

**Data file copies** – Default location for RMAN copies

**RMAN backups** – Default location of RMAN backup/restore
Flash Recovery Area Management

- Disk limit [100%] is reached and a new file needs to be written into the Recovery Area.
- Space pressure occurs.
- Warning issued to user at 85%.
- Critical Alert issued at 97%.
- Oracle deletes files that are no longer required on disk.

- RMAN updates list of files that may be deleted.
- Archivelog & Database file backup.

- Backup files to be deleted.
- Based on Retention Policy setting.

- Check Alert Log
- Db_checkpoint, outstanding_alert ORA-19809
New Background Process

- LGWR
- ARCH
- RVWR
- Redo Logs
- Archived Logs
- Flash Recovery Area
- Flashback Logs
Flashback Database

- Easy and Faster alternative for point-in-time recovery
- Used for recovering Logical data Corruption or User errors
- Eliminate the need to restore a whole database backup
- Can be used on both Primary as well as Standby database
- Eliminate standby database redo apply delay
- Accessible through SQL as well as RMAN
- Use Flashback Log capturing old versions of changed blocks in Recovery Area
- The performance overhead of enabling Flashback Database is less than 2%.

Like a “Rewind” button for the Database
## Monitoring flashback Database

```sql
SQL> SELECT oldest_flashback_scn, 
       oldest_flashback_time, 
       estimated_flashback_size 
FROM V$FLASHBACK_DATABASE_LOG;

<table>
<thead>
<tr>
<th>OLDEST_FLASHBACK_SCN</th>
<th>OLDEST_FLASHBACK_TIME</th>
<th>ESTIMATED_FLASHBACK_SIZE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1535641</td>
<td>24/03/2004 11:14:15</td>
<td>229171200</td>
</tr>
</tbody>
</table>
```

```sql
SQL> SELECT * FROM V$FLASHBACK_DATABASE_STAT;

<table>
<thead>
<tr>
<th>BEGIN_TIME</th>
<th>END_TIME</th>
<th>FLUSHBK</th>
<th>DB</th>
<th>REDO</th>
<th>ESTIMATED_DISK_SIZE</th>
</tr>
</thead>
<tbody>
<tr>
<td>25/03/04 12:17:02</td>
<td>25/03/04 12:28:48</td>
<td>1966080</td>
<td>3514368</td>
<td>1182208</td>
<td>0</td>
</tr>
<tr>
<td>25/03/04 11:16:56</td>
<td>25/03/04 12:16:02</td>
<td>12132352</td>
<td>23576576</td>
<td>8134144</td>
<td>22907289</td>
</tr>
<tr>
<td>25/03/04 10:16:51</td>
<td>25/03/04 11:16:56</td>
<td>12042240</td>
<td>23502848</td>
<td>8310272</td>
<td>227794944</td>
</tr>
</tbody>
</table>
```
BEGIN STEP --- SQL> SHUTDOWN IMMEDIATE
SQL> STARTUP MOUNT;

RMAN> FLASHBACK DATABASE TO SCN=1111111;

SQL> FLASHBACK DATABASE TO TIMESTAMP(SYSDATE -1);

END STEP -- SQL> ALTER DATABASE OPEN RESETLOGS;
Flashback Database Errors

```
SQL> FLASHBACK DATABASE TO SCN 111111;
ERROR AT Line 1:
ORA-38729 : Not enough flashback database log data to do FLASHBACK
```

```
SQL> FLASHBACK DATABASE TO SCN 111111;
ERROR AT Line 1:
ORA-38753 : Cannot FLASHBACK data file 6; no flashback log data.
```

✓ Take the affected datafiles offline and flashback the remaining datafiles.
✓ Recover offline datafiles using Point-in-time recovery methods
Optimizing Usage of FRA

- Identify the Tablespace that are not critical to the business
- Exclude the Tablespace from Flashback Logs

SQL> ALTER TABLESPACE SAMPLEDATA FLASHBACK OFF;

- You need to turn OFF the tablespace FLASHBACK mode after re-creating the control file as this will put all tablespace in FLASHBACK ON mode.

SQL> SELECT TABLESPACE_NAME, FLASHBACK_ON FROM V$TABLESPACE;
Disabling Flashback Database

Disabling Flashback Database will delete all existing Flashback logs

SQL> ALTER DATABASE FLASHBACK OFF;

Or

RMAN> SQL “ALTER DATABASE FLASHBACK OFF”;

Disabling Flashback Database will delete all existing Flashback logs
Flashback DB Benefits [Time]

Incomplete Recovery

In the diagram:
- The process starts with an Oracle 10g database.
- Data files are restored from an archived file.
- Flashback capability allows for recovery.
- The restored data is then used to recover the database to a specific point in time.

DSI - Data Softech Inc.
Complete Database Solution Provider
Flashback DB Benefits [No Delay]

Primary-Standby

Standby 1

Standby 2

Redo/Archived files

No Delay

2 hr Delay

Real-Time Apply

LGWR

RFS

LSP/MRP

Archived Logs

Standby Redo Logs

Archived Logs
Flashback DB Benefits [Failover]

1. Primary-standby Setup
   - Clients
   - Primary: A
   - Standby: B
   - Redo

2. Standby Not Available
   - XX
   - Primary: B
   - Redo

3. Standby Reinstatement
   - XXX
   - Primary: B
   - Redo

4. Primary – Standby Setup
   - Primary: A
   - Standby: B
   - Redo

Flashback DB

1. Primary-standby Setup
   - Clients
   - Primary: A
   - Standby: B
   - Redo

2. Standby Not Available
   - XX
   - Primary: B
   - Redo

3. Standby Reinstatement
   - XXX
   - Primary: B
   - Redo

Flashback DB
Flashback DB Benefits [Resetlogs]

1. Primary-standby Setup
   - Clients
   - Primary
   - Standby
   - Redo

2. Standby Invalidated
   - Resetlogs
   - Primary
   - Standby

3. Standby Reinstatiation
   - Flashback DB
   - Primary
   - Standby

4. Primary – Standby Setup
   - Primary
   - Standby
   - Redo
Flashback Database Restriction

- Cannot used for Media Failure
  - Physical Corruption
  - Deletion of datafiles
- Cannot use it if the Controlfile is restored from backup.
- Cannot used to undo Datafile Shrink operation
- Cannot go beyond the oldest_scn recorded in Flashback logs

```
SQL> SELECT oldest_flashback_scn,
          oldest_flashback_time
       FROM V$FLASHBACK_DATABASE_LOG;
```

- No guarantee if the FRA is not enough.
ORACLE 10g Flashback DROP

Topics
Recycle bin
Flashback Drop

DSI Data Softech Inc.
Complete Database Solution Provider
Flashback DROP

DROP TABLE EMP PURGE

GONE

PURGE
RECYCLEBIN

Drop Table

EMP TABLE

FLASBACK TABLE EMP TO BEFORE DROP

Recycle bin
### RECYCLE BIN

- Virtual or Logical container for dropped table and its dependent objects like:
  - Constraints, triggers, indexes,
  - Lob segment, lob index segments
- No fixed amount of space is allocated to recycle bin
- Dropped Object remain in same tablespace with same Space usage
- Database objects are purged as FIFO
Database objects can purged from recycle bin
- If the corresponding tablespace need new extent
- If the user Quota in the tablespace is exhausted
- For AutoExtendable tablespace, before the datafiles are extended

Dependent objects such as indexes are purged first to meet space pressure
- If all segments of an object are not reclaimed, they are marked as temporary segments and claimed first during next space pressure
- These object are no longer available for Flashback DROP
RECYCLE BIN

- Object names in the RECYCLE BIN are globally unique and are formed as

  BIN$$globalUID$version

  where:
  - globalUID - globally unique, 24 character long identifier
  - version - version number assigned by the database

- The recycle bin name of an object is always 30 characters long.
Perform Recovery: Type

Select Tables

Current Database Information
1. ARCHIVELOG Mode - ARCHIVED
2. Current Status - OPEN

Type
Object Type: Tables

Operation Type
- Recover to current time or a previous point-in-time
  Datafile will be restored as required.
- Restore datafiles
  Need to specify Time, SCN or log sequence. The backup taken at
  or prior to that time will be used.
- Recover from previously restored datafiles
- Block Recovery

Overview
- Restore and/or recover the entire database
  or selected objects
- Restore files to a new location
- Recover tablespaces to a point-in-time
  based on a timestamp, system change
  number (SCN), or log sequence number
- Recover datafile data blocks that are
  marked as corrupted, or based on datafile
  block IDs or tablespace block addresses
Perform Recovery: Dropped Objects Selection

Select the tables from the Recycle Bin that you would like to recover. The Results table shows dependent objects that will also be recovered when the selected tables are recovered.

**Search**

Schema Name: SCOTT

**Results**

<table>
<thead>
<tr>
<th>Select</th>
<th>Object Name</th>
<th>Type</th>
<th>Tablespace</th>
<th>Drop Time</th>
<th>Create Time</th>
<th>Size</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Recycle Bin</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>View Content</td>
</tr>
</tbody>
</table>
Can

Rename

Table Owner | Table Name | New Name
---------------|------------|--------
SCOTT         | DEPT       | DEPT   

Perform Recovery: Rename

Object Type: Tables
Operation Type: Flashback Dropped Tables

Specify a new name for the dropped tables.
Confirmation

The selected tables, SCOTT.DEPT, have been flashed back from the recycle bin.
Table Recovery using Flashback

SQL> SELECT * FROM RECYCLEBIN;
SQL> SHOW RECYCLEBIN

<table>
<thead>
<tr>
<th>ORIGINAL NAME</th>
<th>RECYCLEBIN NAME</th>
<th>OBJECT TYPE</th>
<th>DROP TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEPT</td>
<td>BIN$1lDRxSz0P8XgNAgAIJ3Pqw==$0</td>
<td>TABLE</td>
<td>2004-03-23 2:34:00</td>
</tr>
</tbody>
</table>

SQL> FLASHBACK TABLE "BIN$1lDRxSz0P8XgNAgAIJ3Pqw==$0" TO BEFORE DROP RENAME TO DEPT_INDY;

SQL> SELECT * FROM "BIN$1lDRxSz0P8XgNAgAIJ3Pqw==$0";
Dropping Object Multiple Times

SQL> CREATE TABLE EMP ( ...columns ); # EMP version 1
SQL> DROP TABLE EMP;
SQL> CREATE TABLE EMP ( ...columns ); # EMP version 2
SQL> DROP TABLE EMP;
SQL> CREATE TABLE EMP ( ...columns ); # EMP version 3
SQL> DROP TABLE EMP;

SQL> FLASHBACK TABLE EMP TO BEFORE DROP;
Most recently dropped is retrieved from the recycle bin

SQL> FLASHBACK TABLE EMP TO BEFORE DROP RENAME TO EMP_VERSION_3;
SQL> FLASHBACK TABLE EMP TO BEFORE DROP RENAME TO EMP_VERSION_2;
SQL> FLASHBACK TABLE EMP TO BEFORE DROP RENAME TO EMP_VERSION_1;
Dropping Object Multiple Times

```sql
SQL> CREATE TABLE EMP ( ...columns ); # EMP version 1
SQL> DROP TABLE EMP;
SQL> CREATE TABLE EMP ( ...columns ); # EMP version 2
SQL> DROP TABLE EMP;
SQL> CREATE TABLE EMP ( ...columns ); # EMP version 3
SQL> DROP TABLE EMP;
```

Purge behave differently than Flashback Drop as the First dropped will be the first to be purged

```sql
SQL>PURGE TABLE EMP; # version 1 of able is purged
SQL>PURGE TABLE EMP; # version 2 of table is purged
SQL>PURGE TABLE EMP; # version 3 of table is purged
```
More Purge Options

Purge Index to avoid space pressure
SQL> PURGE INDEX “BIN$1lDRxSz0P8XgNAgAIJ3Pqw===$0”

Purge All dropped objects in tablespace
SQL> PURGE TABLESPACE SAMPLEDATA;

Purge All dropped objects of user SCOTT
SQL> PURGE TABLESPACE SAMPLEDATA USER scott;

Purge All Objects from Recycle bin in the database
SQL> PURGE DBA_RECYCLEBIN;
ORACLE 10g Flashback TABLE
Flashback Table

- Recover a table or tables to a specific point in time without
  - Restoring a backup
  - Taking any part of database offline
  - Without DBA intervention
- Restore table attributes - indexes, constraints, triggers etc
- Flashback to Timestamp or SCN
- Table will be locked during the Flashback operation
Pre-requisite of Flashback Table

- It Use UNDO data stored in UNDO tablespace
- Make sure Enable Row movement on Table

```sql
SQL> ALTER TABLE dept ENABLE ROW MOVEMENT;
```

**Examples**

```sql
SQL> FLASHBACK TABLE employee TO TIMESTAMP TO_TIMESTAMP('2003-04-04 09:30:00', 'YYYY-MM-DD HH24:MI:SS');
```

```sql
SQL> FLASHBACK TABLE employee TO SCN 123456;
```
The execution of the FLASHBACK TABLE statement is recorded in the alert log.

By Default Trigger is disabled for Flashback duration which can be enabled using

```
SQL> FLASHBACK TABLE emp TO TIMESTAMP '2003-03-03 12:05:00' ENABLE TRIGGERS;
```

- This command enables all currently enabled triggers defined on the table.
- You can use ALTER TRIGGER .. DISABLE to selectively disable the triggers before using ENABLE TRIGGERS option. After the completion of above command, enable the disabled trigger.
Perform Recovery: Point-in-time

Object Type: Tables
Operation Type: Flashback Existing Tables

Specify the point in time to which to recover.

- Evaluate row changes and transactions to decide on a point in time
  - Table: SCOTT.EMP
    - Example: SCOTT.EMP

- Flashback to a timestamp
  - Date: Mar 23, 2004
    - Time: 02:14 AM
  - Flashback to a known SCN
    - SCN: 1475895

Time or SCN
Can Add more Tables

SCN

Oracel Enterprise Manager 10g

Perform Recovery: Flashback Tables

Specify the tables you would like to flashback.

Flashback Time: Mar 23, 2004 02:14 PM
Flashback SCN: 1475895

Table Name: SCOTT.EMP

Example: scott.emp, one table name per row

Return to Recovery Type Selection
Perform Recovery: Review

Object Type: Tables
Operation Type: Flashback Existing Tables

The following tables will be flashed back. All these tables will be locked while the flashback operation is in progress.

SCN: 1475895
Timestamp: Mar 23, 2004 02:14 PM
Tables: SCOTT.EMP
Dependent Tables:

Return to Recovery Type Selection
Error Out

Perform Recovery: Review

Error - ORA-08189: cannot flashback the table because row movement is not enabled

Object Type: Tables
Operation Type: Flashback Existing Tables
The following tables will be flashed back. All these tables will be locked while the flashback operation:

<table>
<thead>
<tr>
<th>SCN</th>
<th>Timestamp</th>
<th>Tables</th>
</tr>
</thead>
<tbody>
<tr>
<td>1475895</td>
<td>Mar 23, 2004 02:14 PM</td>
<td>SCOTT.EMP</td>
</tr>
</tbody>
</table>

Dependent Tables
Confirmation

The selected tables, SCOTT.EMP, have been flashed back.
Common Errors - Flashback Table

SQL> FLASHBACK TABLE EMP TO SCN 11111;
ERROR at line 1:
ORA-08183 : Flashback cannot be enabled in the middle of a transaction

Issue COMMIT or ROLLBACK before using FLASHBACK TABLE command
Common Errors - Flashback Table

SQL> FLASHBACK TABLE EMP TO SCN 11111;
ERROR at line 1:
ORA-01466: Unable to read data – table definition has changed

Flashback operation cannot be completed due to following:

- Change the Table Structure
  - Adding the column
  - Dropping a column
  - Modifying a column
- Adding, Dropping, merging, splitting, truncating a partition with the exception of adding range partition
- Moving or Truncating a Table
ORACLE 10g Flashback QUERY

Topics
Flashback Version Query
Flashback Transaction Query
Flashback Versions Query

- Retrieve different committed version of rows existed during a specified interval.
- Use Undo information [Before image data blocks]
- You cannot go beyond UNDO_RETENTION initialization parameter setting.
- Invoked by using the VERSIONS BETWEEN clause of the SELECT statement
- Useful feature to find out
  - How did this happen
  - When and How data is changed
  - Source of Logical corruption [User, application or transaction ] in the database and correct it
- Enables Developer to debug their code.
Flashback Versions Query

- The `VERSIONS BETWEEN` clause does not change the query plan.
- You cannot use the `VERSIONS BETWEEN` clause when you issue a `SELECT` statement against a view. However, you can use the `VERSIONS BETWEEN` clause in a view definition.
- A `SELECT` statement with the `VERSIONS BETWEEN` clause cannot produce versions of the rows when the table specification [DDL] was changed.
Flashback Versions Query Errors

- Mostly if the UNDO_RETENTION value is small as query returns versions up to the UNDO_RETENTION time only.

ORA-30051: VERSIONS clause not allowed here

ORA-30052: Invalid lower limit snapshot expression lower limit below undo_retention

ORA-30053: Invalid upper limit snapshot expression upper limit greater than read snapshot of the queue

ORA-30054: Invalid upper limit snapshot expression upper limit smaller than the lower limit

ORA-30055: NULL snapshot expression not allowed here
Psuedocolumn

- **VERSIONS_STARTTIME**: Returns the timestamp of the first version of the rows returned by the query.
- **VERSIONS_STARTSCN**: Returns the SCN of the first version of the rows returned by the query.
- **VERSIONS_ENDTIME**: Returns the timestamp of the last version of the rows returned by the query.
- **VERSIONS_ENDSCN**: Returns the SCN of the last version of the rows returned by the query.
- **VERSIONS_XID**: For each version of each row, returns the transaction ID (a RAW number) of the transaction that created that row version.
- **VERSIONS_OPERATION**: For each version of each row, returns a single character representing the operation that caused that row version. The values returned are:
  - I → Insert
  - U → Update
  - D → Delete
```sql
SQL> SELECT deptno, dname, loc, versions_operation, versions_xid, versions_starttime
FROM scott.dept
versions between timestamp minvalue and maxvalue
ORDER BY deptno, versions_starttime
```

<table>
<thead>
<tr>
<th>DEPTNO</th>
<th>DNAME</th>
<th>LOC</th>
<th>V</th>
<th>VERSIONS_XID</th>
<th>VERSIONS_STARTTIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>ACCOUNTING</td>
<td>NEW YORK</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>RESEARCH</td>
<td>DALLAS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>SALES</td>
<td>CHICAGO</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>OPERATIONS</td>
<td>BOSTON</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>IT</td>
<td>FRANKFURT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>MARKETING</td>
<td>FRANKFURT</td>
<td>U</td>
<td>1E000A00080000000</td>
<td>02.03.04 09:30:02</td>
</tr>
<tr>
<td>50</td>
<td></td>
<td>FRANKFURT</td>
<td>U</td>
<td>1E000B00080000000</td>
<td>02.03.04 09:31:15</td>
</tr>
<tr>
<td>50</td>
<td></td>
<td>FRANKFURT</td>
<td>D</td>
<td>1E000C00080000000</td>
<td>02.03.04 09:32:20</td>
</tr>
</tbody>
</table>
SQL> SELECT deptno, dname FROM dept 2> VERSIONS BETWEEN TIMESTAMP minvalue AND maxvalue 3> WHERE deptno<20;

<table>
<thead>
<tr>
<th>DEPTNO</th>
<th>DNAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>BU</td>
</tr>
<tr>
<td>1</td>
<td>BU</td>
</tr>
<tr>
<td>10</td>
<td>ACCOUNTING</td>
</tr>
</tbody>
</table>

SQL> SELECT deptno, dname, versions_operation o, 2> versions_xid, versions_starttime FROM dept 3> VERSIONS BETWEEN TIMESTAMP minvalue AND maxvalue 4> WHERE deptno<20;

<table>
<thead>
<tr>
<th>DEPTNO</th>
<th>DNAME</th>
<th>O</th>
<th>VERSIONS_XID</th>
<th>VERSIONS_STARTTIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>BU</td>
<td>U</td>
<td>00020017000004A22</td>
<td>21-OCT-03 11.02.56 PM</td>
</tr>
<tr>
<td>1</td>
<td>BU</td>
<td>I</td>
<td>00020019000004A1A</td>
<td>21-OCT-03 10.52.52 PM</td>
</tr>
<tr>
<td>10</td>
<td>ACCOUNTING</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Enables to view database changes at transaction level
  ➢ Single Transaction
  ➢ All Transactions during a period of time

It is query on view
FLASHBACK_TRANSACTION_QUERY

 Allows you to generate SQL code to undo changes made by any transaction(s).

 Allows fast recovery than Logminer.

 Used to perform audit of transactions.

 Use UNDO data from UNDO tablespace
SQL>
1  SELECT  undo_sql
2  FROM    flashback_transaction_query
3  WHERE   table_owner = 'SCOTT'
4  AND     table_name = 'DEPT'
5  ORDER BY commit_timestamp desc

UNDO_SQL

insert into "SCOTT"."DEPT"("DEPTNO","DNAME","LOC") values ('50','UUuups','Frankfurt');
update "SCOTT"."DEPT" set "DNAME" = 'MARKETING' where ROWID = 'AAALy6AEEAAAAO0AAA';
update "SCOTT"."DEPT" set "DNAME" = 'IT' where ROWID = 'AAALy6AEEAAAA00AAA';
SQL> SELECT logon_user, table_name,
       table_owner, undo_sql
FROM flashback_transaction_query
WHERE table_owner='SCOTT'
AND xid='1E000C0008000000';

LOGON_USER    TABLE_NAME    TABLE_OWNER    UNDO_SQL
-------------- --------------- --------------- ---------------
SMUELLER      DEPT           SCOTT            

insert into "SCOTT"."DEPT"("DEPTNO","DNAME","LOC") values
(50,'UUuuuups','Frankfurt');
<table>
<thead>
<tr>
<th>Object Level</th>
<th>Scenario</th>
<th>Flashback Technology</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>Drop User</td>
<td>Flashback Database</td>
<td>Use Flashback Recovery area</td>
</tr>
<tr>
<td></td>
<td>Batch Job error out</td>
<td>Flashback Database</td>
<td>Use Flashback Recovery area</td>
</tr>
<tr>
<td>Table</td>
<td>Drop TABLE</td>
<td>Flashback Drop</td>
<td>Use Recyclebin</td>
</tr>
<tr>
<td></td>
<td>Truncate TABLE</td>
<td>Flashback Database</td>
<td>Use Flashback Recovery area</td>
</tr>
<tr>
<td></td>
<td>Update table w/o proper Where clause</td>
<td>Flashback Table</td>
<td>Use Undo Segments</td>
</tr>
<tr>
<td></td>
<td>Recover Deleted Data</td>
<td>Flashback Query or Table</td>
<td>Use Undo Segments</td>
</tr>
<tr>
<td></td>
<td>Compare Current data with Past data</td>
<td>Flashback query</td>
<td>Use Undo Segments</td>
</tr>
<tr>
<td>Transactions</td>
<td>Batch job runs twice but not sure what objects affected</td>
<td>Flashback version query &amp; Flashback transaction query</td>
<td>Use Undo Segments</td>
</tr>
</tbody>
</table>
ORACLE 10g/9i DATA GUARD – LOGICAL STANDBY DATABASE

Inderpal S. Johal
Principal Consultant
Data Softech Inc.