

Oracle9i Flashback Query

An Oracle White Paper
March 2002

THE HUMAN ERROR CHALLENGE

Human errors are one of the predominant causes of system failure. According to many studies, about 40% of application outages are caused by operator or user errors. These errors are extremely difficult to avoid and can be particularly difficult to recover from without advance planning and the right technology. Such errors can result in “logical” data corruption, or cause downtime of one or more components of the IT infrastructure. While it is relatively simple to rectify the failure of an individual component, detection and repair of logical data corruption, such as accidental deletion of valuable data, is a time consuming operation which causes enormous loss of business productivity. Most database products today do not offer any solutions to this problem leaving administrators with no option but to recover the database using a previous backup. This not only requires the database to be unavailable during the period of recovery but also leads to the loss of valid transactions if the database is required to be restored to a point in time in the past.

INTRODUCING THE FLASHBACK QUERY

Oracle9i introduces Flashback Query to provide a simple, powerful and completely non-disruptive mechanism for recovering from human errors. It allows users to view the state of data at a point in time in the past without requiring any structural changes to the database. Using Flashback Query, users can execute queries against a snapshot of data consistent to a point in time in the past either by specifying a wall clock time or a System Change Number (SCN). This capability allows users to correct logical errors, or simply view an old image of data, without any disruption in database services. Since Flashback Query can be executed without any administrator intervention, application developers can use this feature to build applications equipped with built in self-service error correction capability. Table data can be queried as it existed at a point in time. This lets applications query, compare, or recover past data without involving the DBA and without an expensive recovery operation. The current table data remains available to other applications throughout.

HOW DOES THE FLASHBACK QUERY WORK?

The Oracle database uses a version based read consistency mechanism by saving an image of the data prior to making any modifications. These changes are stored in the undo tablespace (or Rollback Segments). Should the user later decide not to commit the transaction or if the operation was not successful, Oracle uses the data saved in the undo tablespace to recreate the original data. The undo data is also used to generate a snapshot of data consistent to a point in time when a query was started if another user has modified the underlying data. Flashback Query relies on the same mechanism to construct an image of the data as it existed at a time in the past. The undo data, therefore, must be available in order for a Flashback Query to be successful. Oracle9i's Automatic Undo Management feature allows administrators to specify how long they wish to retain the undo data using the UNDO_RETENTION initialization parameter. By using this parameter and sizing the undo tablespace appropriately, DBAs can control how far back a Flashback Query can go.

USING THE FLASHBACK QUERY

Recovering Accidentally Deleted Data

Having explained the concepts behind the Flashback Query, let us now examine how it can be used in a real world environment. Let us take the example of a banking application where a set of bank accounts was accidentally deleted from the ACCOUNTS table at 11 AM. The inconsistencies caused by this mistake were later discovered at 2 PM. If the bank decides to rollback the database just prior to 11 AM, when the error occurred, in order to "recover" the deleted data, it would lose all subsequently committed transactions. Since rolling back the database requires restoring a backup copy and rolling it forward till the desired point in time, it will also lead to the system being unavailable for the duration of recovery. Clearly, this is not the solution the bank is looking for since it forces the bank to compromise system availability and data integrity.

Using Flashback Query, the database administrator can execute a simple SQL command to recover from such an error. The following query retrieves the deleted accounts by comparing the current data in the ACCOUNTS table against what it was just prior to 11 AM and, inserts them back into the table.

```
INSERT INTO accounts
  (SELECT * FROM accounts AS OF TIMESTAMP
   TO_TIMESTAMP('13-MAR-02 10:59:58', 'DD-MON-YY
   HH24:MI:SS')
   MINUS
   SELECT * FROM accounts);
```

Flashback Query, therefore, makes it possible for the bank to correct this mistake without sacrificing any data and with absolutely no affect on the normal

The SELECT ...AS OF command is only available in Oracle9i Database Release 2. In Oracle9i Database Release 1, the functionality illustrated here can be accomplished by using the DBMS_FLASHBACK procedure.

operations. This enables the bank to avoid any revenue loss due to “lost” data and application outage.

Having realized the immense benefits of using the Flashback Query, the bank now decides to use it to provide self-service error correction capability to its employees. The application is, therefore, enhanced to provide a new menu option that allows users to view the past state of any account and use this data to correct any mistakes that they may have made. The bank employees can now use this feature to “undo” any accidental changes to customer accounts by just using a few mouse clicks.

Tracking Data Changes

While the Flashback Query provides an extremely powerful mechanism of recovering from human errors, its use goes beyond undoing incorrect changes. It can also be used to simplify a number of frequently performed daily operations such as tracking incremental data changes and querying the past state of data.

In the context of the banking application example mentioned earlier, therefore, Flashback Query could also be used to generate a daily report of all new accounts created on any given day by creating a view as shown below.

```
CREATE VIEW accounts_created_today AS
  SELECT * FROM accounts
  MINUS
  SELECT * FROM accounts AS OF TIMESTAMP
  TRUNC(SYSDATE) ;
```

Once this view is created, a list of all accounts created during the day can be produced by executing the following simple query.

```
SELECT * FROM accounts_created_today;
```

Similarly, in order to find out an account balance on a certain day in the past, one simply needs execute a query similar to one shown below.

```
SELECT balance FROM accounts AS OF TIMESTAMP
TRUNC(SYSDATE - 3) WHERE account_number = 12345;
```

The Flashback Query, therefore, simplifies the application design significantly by removing the need to store temporal data. Unlike other database features available for recovering lost data, Flashback Query allows users to view the data as it was in the past and then, lets them choose how to process it. They may wish to analyze the data to determine what exactly caused the change, reverse the change or, simply capture the changed data for further processing.

SECURITY AND PERFORMANCE

Since Flashback Query allows users a new capability to go back in time, administrators are provided control on who can use this feature. In Oracle9i Database Release 1, only users with execute privilege on the DBMS_FLASHBACK procedure can view the past snapshots of data. Oracle9i Database Release 2 provides a more granular control mechanism by introducing the new 'FLASHBACK' object privilege. Consequently, in Oracle9i Database Release 2, users can execute Flashback Query only on those tables for which they have been granted the 'FLASHBACK' privilege. Alternatively, the 'FLASHBACK ANY TABLE' system privilege allows a user to use this feature on any table in the database, with exception of the data dictionary tables. The 'FLASHBACK ANY TABLE' system privilege is included in the DBA role.

The Flashback Query uses the standard undo data storage and read consistency infrastructure in the database. Its use, therefore, does not cause any adverse impact on overall system performance.

CONCLUSIONS

The Flashback Query enables recovery of an entirely new class of errors, those committed by users, with little or no intervention from database administrators. It provides an online mechanism to recover from such errors thereby enhancing system availability and business profitability. In addition, it introduces an extremely powerful functionality in the database to view and process the past snapshots of data, eliminating the need to store it in the application schema. The simplicity of this feature allows application developers to easily build new functionalities using the past versions of data and provide application level, end user self-service error correction capabilities. By introducing a standard infrastructure for application developer to access the past data and liberating database administrators from the task of correcting users errors, Flashback Query enhances their productivity and, reduces the application development and deployment costs significantly.



White Paper Title: Oracle9i Flashback Query
August 2001

Author: Sushil Kumar, Tammy Bednar

Contributing Authors:

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:

Phone: +1.650.506.7000

Fax: +1.650.506.7200

www.oracle.com

Oracle Corporation provides the software
that powers the internet.

Oracle is a registered trademark of Oracle Corporation. Various
product and service names referenced herein may be trademarks
of Oracle Corporation. All other product and service names
mentioned may be trademarks of their respective owners.

Copyright © 2000 Oracle Corporation
All rights reserved.