An Oracle White Paper
September 2009

Oracle Total Recall with Oracle Database 11g Release 2

Introduction: Total Recall = Total History

In a perfect world, organizations would keep track of every change to every piece of data they manage. However, keeping track of history is complicated: extensive application logic is needed, and everything has to be kept in sync as file and database definitions change. Viewing history is also complicated: querying and reporting on the data is hard, and viewing data across multiple history tables is even more difficult. As a result, organizations spend huge amounts of time and money making sure their applications correctly track history for a small number of key fact tables, such as sales, payments, customers, etc., and leave the rest of their data without full contextual history. The Oracle Total Recall option makes it trivial to securely track and query historical data for any database table, without the cost or performance impact of traditional approaches.

Managing Historical Data: Current Approaches

There are several traditional methods for tracking historical data; all of these methods suffer from a combination of high implementation costs, high maintenance costs, performance limitations, and complexity.

Application logic

One approach to implementing historical data management is at the application level: the logic for history management and archiving is embedded within the business logic. A key advantage of this approach is that applications understand the business context and can provide easy grouping of different business components that need to be tracked. However, this comes with a cost: increased application complexity. Data access has to be implemented through defined application interfaces to ensure accurate and complete history tracking. Direct updates to the data at the database level could introduce data integrity issues with the history, and could invalidate the entire set of historical data.

In an organization where there are typically hundreds of applications, with a mix of third party and homegrown applications, implementing historical data management in every application would be extremely difficult. This problem is further exacerbated by application upgrades, changing retention requirements, and the lack of a centralized management interface.

Database triggers

An alternative approach is to use database triggers to maintain data history. This approach can be implemented without changing the related applications. However, using triggers doesn't provide a centralized management interface, and can have a significant impact on application performance. In addition, privileged users can modify the historical data, calling into question the integrity of that data.

Redo log mining

Another approach is to mine redo logs to create and store data history. Log mining requires a separate set of tools and processes to configure, monitor, and manage. As with application logic and database triggers, privileged users can modify the historical data stored by log mining, calling into question the integrity of that data.

None of these approaches provide a low-cost, high-performance, secure, and seamless history data management system.

Flashback Data Archive

The Flashback Data Archive (FDA) feature of the Oracle Total Recall option in Oracle Database 11g removes the cost, performance, and complexity barriers to securely tracking the contextual history of all data. Flashback Data Archive makes it possible to automatically and transparently track all of the changes to any set of tables in an Oracle 11g database, and to easily query data in those tables as of any point in time or over any interval within the specified retention period, with minimal performance impact.

A Tool for Many Uses

Flashback Data Archive in Oracle Database 11gR2 can be used to solve several distinct business problems. Examples include:

- Data forensics: find and revert changes made by a disgruntled employee

- ILM (Information Lifecycle Management): guarantee immutable history of financial data

- Retention policy enforcement: automatically purge history older than 5 years

- Historical reporting: analyze product changes over time

- Error recovery: restore records that were erroneously removed or updated

- Employee fraud detection: find assets that were deleted but never sold

- Tamper-proof history: retain a SecureFiles file even when the original is destroyed

Using Flashback Data Archive is a simple process: create a tablespace to store the flashback data, create a flashback data archive in the new tablespace, and enable the flashback data archive feature on the tables you want to track.

Benefits

Flashback Data Archive overcomes the limitations inherent in current approaches to historical data management. It automatically tracks every change made to FDA-enabled data stored in the database, and maintains a secure, efficient and easily accessible archive of historical data. The historical data can be retained as long as needed, and is easily accessible with standard SQL statements. Flashback Data Archive can be enabled on both existing and new tables instantaneously and in a completely application transparent manner.

The primary advantages of using Flashback Data Archive for historical data tracking include:

- Application transparency – Enabling Flashback Data Archive on one or more tables can be done instantaneously with no application changes. Customers can use this feature to capture historical data from both packaged and in-house applications.

- Seamless access – Historical data can be easily accessed using familiar Flashback SQL constructs. Flashback Data Archive includes support for flashback queries: applications can seamlessly query the history of data as it existed at any point in time or during any time interval within the retention period. No special snapshots are needed to enable this feature.

- Security – historical data, once generated, is immutable to all users. Access to the internal history tables is restricted to reads only. No DML operations on historical data are allowed for any users, including administrators. Applications need not query the internal history tables directly as seamless access is provided through the Flashback Query mechanism.

- Minimal performance overhead – Regular user transactions will see negligible impact. Flashback Data Archive employs a lightweight mechanism to mark DML operations on FDA-enabled tables. The actual history generation and archiving is done asynchronously with background processes.

- Storage optimization –Flashback Data Archive compresses and partitions the internal history tables to optimize storage and performance. Both compression and partitioning in Flashback Data Archive are managed automatically and require no special administration.

- Centralized management – Flashback Data Archive provides a centralized and policy-based management interface to automate ongoing administrative tasks. With Flashback Data Archive, you can easily group tables and set a common retention policy for each group. When a new table is added to a flashback data archive, it automatically inherits the retention parameter from that archive. The Oracle database automatically purges aged-out history data for all tracked tables based on the specified retention policy. This frees up the administrator from the repetitive management of history data and avoids costly errors associated with manual maintenance such as purging the wrong history. Regulatory compliance is simplified by limiting management of historical data to a small user group.

Understanding Flashback Data Archive

A flashback data archive is a logical container for managing historical information for specified tables. It is a new dictionary object in Oracle Database 11g that defines archive storage and data purging policies. A flashback data archive can span multiple tablespaces. Administrators can define the amount of space a flashback data archive can use in each tablespace using the 'QUOTA' parameter. Multiple flashback data archives can be created as needed to implement different archiving policies. Each flashback data archive contains a 'RETENTION' parameter that specifies the duration for retaining historical changes. Flashback Data Archive guarantees that historical data will be retained for the duration specified by 'RETENTION' and automatically purges out aged historical data. Additionally, administrators with the necessary privileges can purge historical data in an ad-hoc manner.

Flashback Data Archive creates an internal history table for every tracked table. The internal history table is initially a replica of the FDA-enabled table with additional metadata columns. When one or more columns in the tracked table are updated, a new row is inserted into the history table that is the before-image of the row before the transaction. UPDATE and DELETE operations generate a new record in the history table, but INSERT operations do not – inserted rows appear in the base table. The internal history table is partitioned for better performance, and compressed to reduce disk space requirements. No modifications are allowed to internal history tables. Applications and users can use the 'AS OF' and 'VERSIONS BETWEEN' SQL constructs to seamlessly query the historical data.

Architecture

Oracle implements a multi-versioning mechanism that ensures read consistency while maintaining a high degree of concurrency. When DML operations such as INSERT, UPDATE, or DELETE are executed, Oracle writes data to an undo tablespace that is used for transaction rollbacks and for guaranteeing read consistency.

Flashback Data Archive works with undo data as do most other Flashback features. However, the historical data retained in the flashback data archive is not limited by the size of the undo tablespace. History generation is implemented in a non-intrusive manner through a new background process called FBDA. After a table has been enabled for history tracking with Flashback Data Archive, all transactions on the table and the corresponding undo records are marked for archival. In order to guarantee that every such transaction is archived, the undo records are not recycled until the history is generated and stored in the database.

The FBDA process sleeps and wakes up at system-managed intervals and processes the undo data marked for archival. After 'fbda' generates the history, the transactions and undo records are candidates for recycling. The asynchronous architecture of Flashback Data Archive ensures a negligible impact on the overall system performance.

Flashback Data Archive architecture
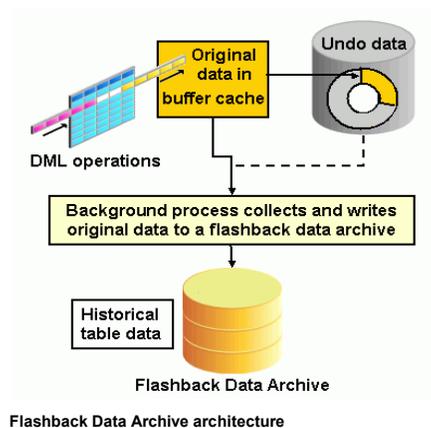
The 'fbda' process intelligently adjusts its sleep interval based on the system undo generation rate. As transaction activity increases, 'fbda' automatically reduces its sleep interval from the default of 5 minutes. For better performance, 'fbda' also adjusts its sleep time in order to maximize undo data reads from the buffer cache. In addition, Flashback Data Archive uses an internal partitioning scheme for all the historical data for better peformance, and it uses table compression to reduce the storage footprint of the historical data.

Flashback Data Archive Example

The following simple example is designed to show how to enable and use Flashback Data Archive.

Flashback Data Archive Requirements

1. Flashback Data Archive tablespaces must be managed with automatic segment space management (ASSM).

2. Automatic Undo Management must be enabled.

Step-by-Step Approach

There are 3 steps to enabling history tracking on tables using Flashback Data Archive:

1. Either create a new tablespace or use an existing one. The flashback archive tablespace must be managed with automatic segment space management (ASSM). Optionally, it can also have Automatic Storage Management (ASM) enabled to improve performance.

2. Create a flashback data archive:

CREATE FLASHBACK ARCHIVE fda1
TABLESPACE tbs1
RETENTION 5 YEAR;

This creates a new flashback data archive called 'fda1' on tablespace tbs1 with a retention policy of 5 years. Since 'QUOTA' is not specified, the default setting of 'UNLIMITED' is used. At this point, we have created a logical container for historical data.

- The FLASHBACK ARCHIVE ADMINISTER system privilege is required for creating a new flashback data archive

- The following static data dictionary views are available

  - DBA/USER_FLASHBACK_ARCHIVE – Displays information about flashback data archives

  - DBA/USER_FLASHBACK_ARCHIVE_TS – Displays tablespaces and the mapping to flashback data archives

3. Enable Flashback Data Archive on the desired tables.

ALTER TABLE EMPLOYEES FLASHBACK ARCHIVE fda1;

The EMPLOYEES table is now enabled for Flashback Data Archive, and its history data will be stored in the flashback data archive 'fda1'. When the first DML operation occurs, the internal history table is created in 'fda1'.

- The FLASHBACK ARCHIVE object privilege is required to enable Flashback Data Archive

- The following static data dictionary views are available:

  - DBA/USER_FLASHBACK_ARCHIVE_TABLES – Displays information about tables that are enabled for Flashback Data Archive

Querying History Data

Flashback Data Archive provides seamless access to the historical data using the 'AS OF' or 'VERSIONS BETWEEN' SQL constructs. You can query for the state of any row in the tracked table as far back as your specified retention period.

The following is an example for querying the salary details for the employee with id=193 on June 1, 2007:

```
SELECT last_name, first_name, salary
FROM EMPLOYEES
AS OF TIMESTAMP TO_TIMESTAMP('2007-06-01 00:00:00','YYYY-MM-DD HH24:MI:SS')
WHERE employee_id=193;
```

New in Oracle Database 11g Release 2

Versions Query

The following is an example for querying the salary details for the employee with id=193 during the two years from June 1, 2007 to June 1, 2009:

```
SELECT last_name, first_name, salary
FROM EMPLOYEES
VERSIONS BETWEEN TIMESTAMP
TO_TIMESTAMP('2007-06-01 00:00:00','YYYY-MM-DD HH24:MI:SS') AND
TO_TIMESTAMP('2009-06-01 00:00:00','YYYY-MM-DD HH24:MI:SS')
WHERE employee_id=193;
```

Schema Evolution

Flashback Data Archive guarantees historical data capture for all transactional changes to tracked tables. In Oracle Database 11g Release 1, the Add Column DDL operation is supported with Flashback Data Archive. With Oracle Database 11g Release 2, the following DDL operations are supported, with full support for Flashback queries across all associated changes:

- Add, Drop, Rename, Modify Column

- Drop, Truncate Partition

- Rename, Truncate, Drop Table

Disassociate / associate

For more complex DDL – upgrades, split table, etc. – the Disassociate and Associate PL/SQL procedures can be used to temporarily disable Flashback Data Archive on specified tables. The Associate procedure enforces schema integrity after association: the base table and history table schemas must be the same. The Disassociate and Associate procedures require the FLASHBACK ARCHIVE ADMINISTER privilege.

Performance Enhancements

In Oracle Total Recall for Oracle Database 11g Release 2, several performance enhancements have been added to ensure that Flashback Data Archive has a negligible impact on performance.

- Multiple FBDA processes during archiving: the FBDA process can spawn multiple slaves

- Parallel DML: the FBDA process and its slaves can execute parallel DML

- Auto-tuning for heavy loads: the FBDA process will automatically spawn slaves and shorten its sleep time based on increasing demand for archiving throughput

- Bulk archiving of small transactions: the FBDA process bundles smaller transactions to increase throughput

In tests using the TPC-C standard benchmark, insurance data, and internal benchmarks, the average response time impact with Flashback Data Archive enabled was about 5%, compared to an average performance impact with triggers of about 54%.

Privileges and Management of Flashback Data Archive

FLASHBACK ARCHIVE ADMINISTER – a new system privilege managing flashback data archives. Allowed operations include:

- Create new flashback data archive
- Modify existing flashback data archive
  - Make a specific flashback data archive as the default
  - Set RETENTION time
  - Ad-hoc purge of history data
  - Add/remove tablespaces
- Disable tracking for tracked tables
- Drop flashback data archive
- Execute Disassociate and Associate procedures

FLASHBACK ARCHIVE – a new object privilege for enabling flashback data archive on tables

Recommended Best Practices

To optimize performance and space consumption, Flashback Data Archive does not automatically replicate base table indexes on the internal history tables. We recommend that the DBA/USER_FLASHBACK_ARCHIVE_TABLES view be accessed for the name of the internal table for a given tracked table, and the desired indexes (depending on the historical data access patterns of the application) created on the internal table. These indexes will have a performance impact on the Flashback Data Archive process as the database will automatically maintain the indexes for any data inserted, updated, or deleted in the internal tables.

We also recommend the use of dedicated tablespaces for Flashback Data Archives. The QUOTA parameter should be left at the default value of Unlimited. This makes space planning and

maintenance of flashback data archives easier. Also, it is advisable to group tables based on the retention periods and create a separate flashback data archive for each retention period.

Finally, access to the system privilege FLASHBACK ARCHIVE ADMINISTER should be restricted. It is recommended to separate user access to the system privilege FLASHBACK ARCHIVE ADMINISTER from the object privilege FLASHBACK ARCHIVE.

Conclusion

The Flashback Data Archive feature of the Oracle Total Recall option provides a secure, efficient and application-transparent solution for generating and managing historical data. It provides a centralized and integrated interface to manage and retain data history, and automated, policy-based management that greatly simplifies administration. With Flashback Data Archive, you can easily implement historical data tracking to comply with new regulations or adapt to changing business needs.

ORACLE®

White Paper Title
September 2009
Author: Kevin Jernigan
Contributing Authors: Liang Guo, Vasudha
Krishnaswamy, Venkatesh Radhakrishnan,
Vivekanandhan Raja, Tim Shetler

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
oracle.com

Oracle is committed to developing practices and products that help protect the environment