

Warehouse Promotion

Updating Tables with Partition Exchange
Randy Wilcox
March 25, 2014

Potential
of One

Power
of
All



Update your data warehouse

Updates to your data warehouse usually follow one of these methods

- Update tables in place
- Perform updates to copies of existing tables with an alternate name, then rename all tables during promotion to production
- Perform updates to a different library, database or schema

Updates in place

Usually requires exclusive access to the warehouse or each table that is being updated

Do not have read consistency

Updates to copies of existing tables

Usually involves up to 3 copies of each table

- Pre_table_name – This is the table copy to be updated
- Table_name – This is the table copy in production
- Post_table_name – This is the backup that was previously in production

Perform updates in alternate library, database or schema

Requires additional DB resources

Requires LIBNAME assignments to change with production promotion

Promotion via Partition Exchange

The functionality behind exchange partition is built into Oracle Enterprise Edition with the Partitioning Option (additional license).

A table is defined as a partitioned table either via some range or value. The range may include date values, which makes this very useful for transaction data/fact tables.

Exchanges are instantaneous, can be query safe and has statistics and indexes created prior to swap.

Create a partition table

- Create a dimension table
- Create indexes
- Add some records
- Gather statistics

Create a Partition Table

Partition MY_DIM_A will contain all of the data in this example. The MY_DIM_H partition is just a dummy that must exist because a table must have at least two partitions.

```
CREATE TABLE MY_DIM
(
  MY_SK    NUMBER (12),
  MY_NK    NUMBER (12),
  ADDRESS  VARCHAR2 (150),
  LAST_NAME VARCHAR2 (50),
  FIRST_NAME VARCHAR2 (50)
)
PARTITION BY RANGE (MY_SK)
(PARTITION MY_DIM_H VALUES LESS THAN (1),
 PARTITION MY_DIM_A VALUES LESS THAN (MAXVALUE));
```

Create Partition Table Indexes

We create a local partitioned index. This means that each partition of the index is associated with a single partition of the table.

When the exchange occurs the indexes will be exchanged along with the table data.

```
CREATE INDEX MY_DIM_IDX01
ON MY_DIM (MY_SK)
LOCAL (PARTITION MY_DIM_H,
PARTITION MY_DIM_A);
```

Add some records

Add default records

```
insert into MY_DIM values (-1, -1, null, 'UNKOWN', 'UNKNOWN');
```

```
insert into MY_DIM values (0, 0, null, 'INVALID', 'INVALID');
```

Add some additional records

```
insert into MY_DIM values (1, 1, '123 ABC Street', 'Smith', 'Mary');
```

```
insert into MY_DIM values (2, 2, '456 DEF Street', 'Johnson', 'Frank');
```

```
insert into MY_DIM values (3, 3, '789 GHI Street', 'World', 'Hello');
```

Gather Statistics on the Partition Table

Optimizer statistics must be gathered any time the contents of a table change substantially.

We use the “Granularity => ‘ALL’” parameter to ensure that stats are gathered at both the partition and global levels.

```
BEGIN
SYS.DBMS_STATS.GATHER_TABLE_STATS (
  OwnName      => 'WHOUSE'
  ,TabName     => 'MY_DIM'
  ,Estimate_Percent => SYS.DBMS_STATS.AUTO_SAMPLE_SIZE
  ,Method_Opt  => 'FOR ALL COLUMNS SIZE AUTO '
  ,Granularity => 'ALL'
  ,Cascade     => TRUE);
END;
```

Create an exchange table

- Create the table with same format as partition table and create the indexes
- Insert with new data
- Gather statistics

Create a non-partitioned exchange table

This is the table that will be loaded by the ETL process.

Columns, data types, and indexes must exactly match the partitioned table.

```
CREATE TABLE ZZ_MY_DIM
(
  MY_SK    NUMBER (12),
  MY_NK    NUMBER (12),
  ADDRESS  VARCHAR2 (150),
  LAST_NAME VARCHAR2 (50),
  FIRST_NAME VARCHAR2 (50)
);

CREATE INDEX ZZ_MY_DIM_IDX01
ON ZZ_MY_DIM (MY_SK);
```

Add some records

Add some additional records:

```
insert into ZZ_MY_DIM select * from MY_DIM partition(MY_DIM_A);  
insert into ZZ_MY_DIM values (4, 4, '321 CBA Street', 'Anderson', 'Neo');  
insert into ZZ_MY_DIM values (5, 5, '654 FED Street', 'Bond', 'James');  
insert into ZZ_MY_DIM values (6, 6, '987 IHG Street', 'Blues', 'Delta');
```

Gather statistics on the exchange table

Optimizer statistics must be gathered any time the contents of a table change substantially.

These statistics will be exchanged into the partitioned table along with the data and indexes when the swap occurs.

```
BEGIN
SYS.DBMS_STATS.GATHER_TABLE_STATS (
  OwnName      => 'WHOUSE'
  ,TabName     => 'ZZ_MY_DIM'
  ,Estimate_Percent => SYS.DBMS_STATS.AUTO_SAMPLE_SIZE
  ,Method_Opt  => 'FOR ALL COLUMNS SIZE AUTO '
  ,Cascade     => TRUE);
END;
```

Perform the partition exchange

The following statement displays the data, exchanges the data in the ZZ_MY_DIM exchange table for the contents of the active partition of MY_DIM and displays the results

```
select * from MY_DIM;
```

```
select * from ZZ_MY_DIM;
```

```
ALTER TABLE MY_DIM EXCHANGE PARTITION MY_DIM_A WITH TABLE ZZ_MY_DIM INCLUDING INDEXES  
WITH VALIDATION;
```

```
select * from MY_DIM;
```

```
select * from ZZ_MY_DIM;
```

The indexes for ZZ_MY_DIM are exchanged along with the data so they don't need to be rebuilt.

Describe a date based fact/transaction table

- Create a range-based partition table suitable for fact/transactions
 - Range based on transaction date or another date like number such as 20140325
 - Ranges are usually based on months or days
- Review the new exchange table creation
 - New records can be loaded into an exchange table
- Review how to perform a “one way” swap of a new partition
 - A new partition is created in the transaction table.
 - The newly loaded exchange table is swapped into the new, empty partition.

Where to go from here?

There are a number of great resources on the Internet for Oracle Partitioning and the syntax around exchanging partitions. I recommend Oracle's online documentation along with SAS Support documentation regarding methods of interacting with Oracle via SAS.

Research the benefits of partitioning and the methods of exchanging partitions. Even if you choose not to use exchange partitions, Oracle Partitioning can prove very beneficial to query performance, data management including archiving and compression.

Further Reading

Oracle White Paper on partitioning -

<http://www.oracle.com/technetwork/database/enterprise-edition/partitioning-11g-whitepaper-159443.pdf>

Oracle Documentation -

http://docs.oracle.com/cd/E11882_01/server.112/e25523/intro.htm#VLDBG00101

SAS/ACCESS 9.4 for Relational Databases: Reference, Third Edition -

<http://support.sas.com/documentation/onlinedoc/access/index.html>

SAS Solutions OnDemand

Questions?

Contact SAS Solutions OnDemand

- Learn more about our services:
<http://www.sas.com/solutions/ondemand/index.html>
- Email: Randy.Wilcox@sas.com
- Blog: <http://randywilcoxdba.wordpress.com/>



Washington, D.C.
March 23–26, 2014