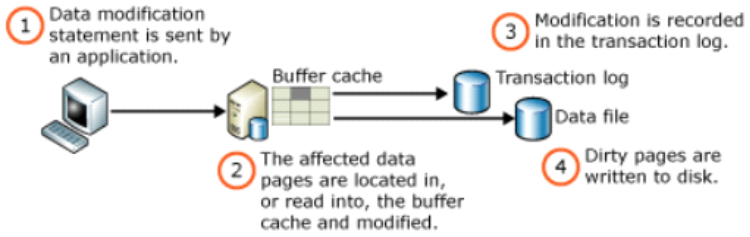
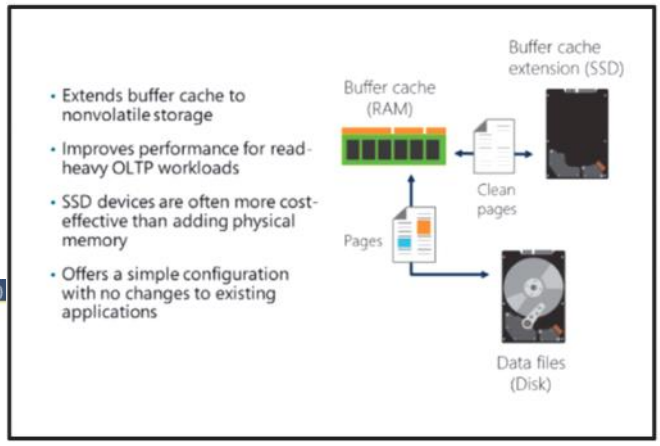
Buffer Pool Extension

Sunday, February 10, 2019 4:38 PM





AlterBufferPool.sql...-OFFICE\dellp (58))* | ColumnsStore.NonC...OFFICE\dellp (59)) | ColumnsStore.Clust...-OFFICE\dellp (58))

```
1  USE MASTER;
2  GO
3  ---------------------------------------------
4  --Add MORE room for Buffering to an SSC
5  ---------------------------------------------
6  ALTER SERVER CONFIGURATION
7      SET BUFFER POOL EXTENSION ON
8      (FILENAME = 'C:\Data\MYCACHE.BPE', SIZE = 50 GB);
9  GO  --.bpe (Buffer Pool Extension)
10 --C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER\MSSQL\DATA
11
12
13 ------------------------------------------------
14 --To reconfigure Turn off then ALTER again
15 ------------------------------------------------
16 --ALTER SERVER CONFIGURATION
17 --  SET BUFFER POOL EXTENSION OFF;
18 --GO
19
20 --ALTER SERVER CONFIGURATION
21 --  SET BUFFER POOL EXTENSION ON
22 --  (FILENAME = 'C:\Data\MYCACHE.BPE', SIZE = 100 GB);
23 --GO
24
```

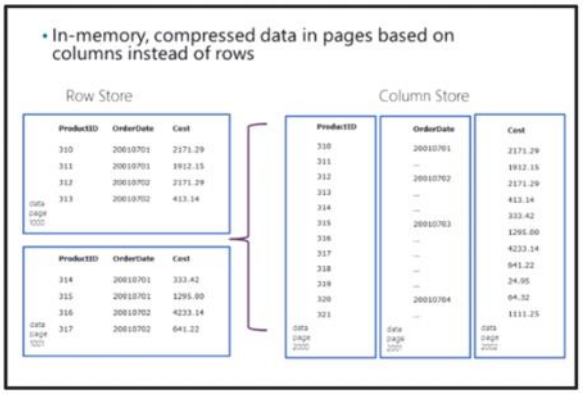
Clustered Column Store Index

Monday, December 4, 2017 10:14 PM

Updating Clustered Columnstore Indexes

You can update clustered columnstore indexes, and you can bulk-load, insert, update, and delete data in a clustered columnstore indexed table by using standard Transact-SQL statements.

- When you use an **INSERT** statement to insert a new row, it is stored in the deltastore until there are enough rows to meet the minimum size for a rowgroup. This rowgroup is then compressed and moved into the columnstore segments.
- When you execute a **DELETE** statement, affected rows that are stored in the deltastore are physically deleted. Affected data in the columnstore segments is marked as deleted and the physical storage is only reclaimed when the index is rebuilt.
- When you execute an **UPDATE** statement, affected rows in the deltastore are updated. Affected rows in the columnstore are marked as deleted and a new row is inserted into the deltastore.



ColumnsStore.NonCl...(LABS\dellp (51)) ColumnsStore.Clust...a (LABS\dellp (54))

```
USE  LabData
GO

CREATE TABLE  dbo.CourseColumnStore (
     ID int NOT NULL,
     CourseCode int NOT NULL,
     Name nvarchar (255) NOT NULL,
     Description nvarchar (max) NOT NULL,
     StartDate date  NOT NULL,
     Age int NULL)
GO

CREATE CLUSTERED COLUMNSTORE INDEX CourseColumnClustered_Index ON dbo.CourseColumnStore
With (Drop_Existing=On);
GO

--DROP TABLE  CourseColumnStore
```

Non-Clustered Column Store Index

Saturday, December 2, 2017 2:19 PM

Nonclustered Columnstore Indexes

A nonclustered columnstore index has the following characteristics:

- It can include some or all of the columns in the table.
- It can be combined with other indexes on the same table.
- You cannot update it. Tables that contain a nonclustered columnstore index are read-only.

ColumnsStore.NonCl...(LABS\dellp (51)) | ColumnsStore.Clust...a (LABS\dellp (54))

```
1  USE  LabData
2  GO
3
4  CREATE TABLE  dbo.CourseColumnStore (
5       ID int NOT NULL,
6       CourseCode int NOT NULL,
7       Name nvarchar (255) NOT NULL,
8       Description nvarchar (max) NOT NULL,
9       StartDate date  NOT NULL,
10      Age int NULL)
11 GO
12
13 CREATE NONCLUSTERED COLUMNSTORE INDEX CourseColumnNonClustered_Index ON dbo.CourseColumnStore
14 (
15      ID,
16      CourseCode,
17      Name,
18      StartDate,
19      Age
20 )
21 GO
22
23 --DROP TABLE  CourseColumnStore
24
```