Columnstore Indexes Worst Practies & Less Known Limitations

by Niko Neugebauer
Niko Neugebauer

Microsoft Data Platform Professional
OH22 (http://www.oh22.net)
Data Platform MVP
Founder of 3 of Portuguese PASS Chapters
Creator of CISL – Columnstore Indexes Script Library
(https://github.com/NikoNeugebauer/CSIL)

Blog: http://www.nikoport.com
Twitter: @NikoNeugebauer
LinkedIn: http://pt.linkedin.com/in/webcaravela
Today’s Agenda is Simple

- Intro
- Overall Recommendation
- Building Columnstore Indexes
- Loading Data into Columnstore Indexes
- Querying the Columnstore Indexes
Columnstore Indexes are:

- Vertically separated
- Grouped into Segments
- Extremely compressed
- Tuned for processing large volumes of data

- A Row Group CAN contain between 1 and 1048576 rows
Today’s Agenda I

Overall:
- Not using the highest available compatibility level

Building
- Cutting on Memory
- Using Strings
- Partitioning
- Compression Delay
Today’s Agenda II

Loading:
- Not using Bulk Load API (<102,400 rows)
- Identity
- Merge
- Parallel Loading
- SSIS
Today’s Agenda III

Querying

- Casting on the wrong side
- Overusing the NULL Expressions
- Focusing on Cross Apply where it is not needed
- Not using the Advanced Storage Engine Features
- Estimated are similar! Similar How?
- Using Natively Compiled Stored Procedures
Identity

SEQUENCE Data Loading Performance (in seconds)

- Single-Threaded: 12 seconds
- Parallel Insertion: 8 seconds
- No Identity or Sequence: 4 seconds
Merge

1 Million Rows Data Loading with
MERGE vs DELETE + INSERT (time in Seconds)

MERGE

DELETE + INSERT
Scaling Merge?

Data Loading with
MERGE vs DELETE + INSERT

- **MERGE**
- **DELETE + INSERT**
- Linear (MERGE)
- Linear (DELETE + INSERT)

1 Million Rows vs 5 Million Rows
Merge Execution Plans

Query 1: Query cost (relative to the batch): 24%
DELETE Target FROM dbo.lineitem_csi as Target WHERE EXISTS | SELECT 1 FROM dbo.lineitem_csi Stage as Source WHERE Target.L_ORDERKEY = Stage.L_ORDERKEY

Query 2: Query cost (relative to the batch): 74%
INSERT INTO dbo.lineitem_csi WITH (TABLOCK) SELECT L_SHIPDATE, L_ORDERKEY, L_DISCOUNT, L_EXTENDEDPRICE, L_SUPPLYKEY, L_QUANTITY,
Thank you very much!
Resources:

My Columnstore Blogpost Series (100+):
http://www.nikoport.com/columnstore

CISL – Open Source Columnstore Library:
https://github.com/NikoNeugebauer/CISL