

I have indexes

But do I have the right indexes?

Eric Blinn

Sr Data Architect

Squire Patton Boggs

Who is this guy?

Sr Data Architect @  SQUIRE
PATTON BOGGS

Vice President @  PASS
LOCAL GROUP

Content Creator @  DRNET
A GYM MEMBERSHIP FOR YOUR CAREER

• Contact info:

- ericblinn.com
- [linkedin.com/in/ericblinn](https://www.linkedin.com/in/ericblinn)
- Twitter: [@SQL2TheSequel](https://twitter.com/SQL2TheSequel)

About this presentation

Technical, but not a deep dive

Intermediate level

Agenda

- Traditional B-Tree indexes
- Full text
- Columnstore

B-Tree Indexes

Clustered and non-clustered indexes

The best value mix of non-clustered indexes

Filtered Indexes

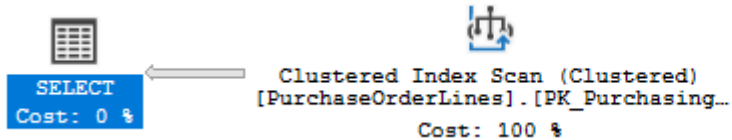
Compressed Indexes

Index usage

Query 1: Query cost (relative to the batch): 100%

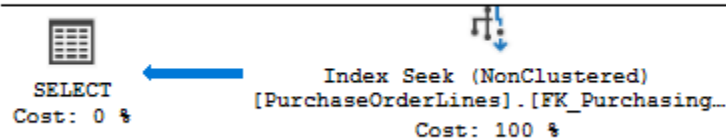
```
SELECT * FROM [Purchasing].[PurchaseOrderLines] WHERE [LastReceiptDate]=@1
```

Missing Index (Impact 97.0562): CREATE NONCLUSTERED INDEX [<Name of Missing



Query 1: Query cost (relative to the batch): 1

```
SELECT [PurchaseOrderLineID],[PackageTypeID] F
```



Best Value

sys_dm_db_missing_index_groups

DEMO

sys_dm_db_index_usage_stats

*Available starting in SQL Server 2005 Standard

Filtered Indexes

Non-clustered index with a subset of rows

Available starting with SQL Server 2008 Standard

```
CREATE INDEX [IndexName] ON Sales.Invoices(ContactPersonID)
WHERE InvoiceDate >= '1/1/2012' AND InvoiceDate <= '1/31/2012'
```

DEMO

Compressed Indexes

Clustered, non-clustered
Compressed at ROW or PAGE level

Available starting with SQL Server 2008 Enterprise

Standard Edition starting with SQL Server 2016 SP1

Compressed Indexes

Only compresses on-page data.
Not BLOBs! Not MAXs!

Reduces data between 40%-60% on average

Stays compressed in buffer pool

Adds about 10% CPU overhead

Compressed Indexes

```
CREATE INDEX MyIndex ON dbo.MyTable  
WITH (DATA_COMPRESSION = PAGE | ROW)
```

DEMO

```
ALTER TABLE dbo.MyTable REBUILD  
WITH (DATA_COMPRESSION = PAGE | ROW)
```

Full Text Indexes

Full text is a companion service to SQL Server

Based on Windows file search algorithm

Search for words within text columns

Available since SQL Server 2000

Starting in 2012 the service added delimiters and more powerful search abilities

Full Text Limitations

Updates are asynchronous

Tables must have single column unique index

Full text indexing will help you find entire words, starts of words, or forms of words in text columns

Full text indexing will **not** help you find **parts** of words.

Full Text Indexes

Consider this list of movies:

ID ^{pk}	Title	Director
1	Wind River	Taylor Sheridan
2	Blade Runner 2049	Denis Villeneuve
3	Baby Driver	Edgar Wright
4	My Cousin Rachel	Roger Michell

Full Text Indexes

Table	docid	Column	Word	Sequence
Movie	1	Title	Wind	1
Movie	1	Title	River	2
Movie	1	Author	Taylor	1
Movie	1	Author	Sheridan	2
Movie	2	Title	Blade	1
Movie	2	Title	Runner	2
Movie	2	Title	2049	3
Movie	2	Author	Denis	1
Movie	2	Author	Villeneuve	2
Movie	3	Title	Baby	1
Movie	3	Title	Driver	2
Movie	3	Author	Edgar	1
Movie	3	Author	Wright	2

FullText index catalog

DEMO

New Keywords

	Exact Matches	Fuzzy Matches
Boolean (WHERE)	CONTAINS	FREETEXT
Table Valued (FROM)	CONTAINSTABLE	FREETEXTTABLE

Columnstore Indexes

Stores data by the column rather than the row

Compression automatically included

Column order doesn't matter

B-Tree Use Case

```
SELECT * FROM dbo.Invoice WHERE CustCode = 'ITDEV'
```

Column1	Column2	Column3	Column4	Column5	Column6	Column7	Column8	Column9
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii
iiiiiiii	iiii	iiii	iiiiiiii	iii	iiiiiiiiii	ii	iiiiiiii	ii

Columnstore Use Case

Big queries

Grouping and aggregation

Lots of reads, limited writes

Columnstore over the years

SQL Server 2012 Enterprise Only

Read-only Non-Clustered Columnstore Index

Msg 35339, Level 16, State 1, Line 1
Multiple nonclustered columnstore indexes are not supported.

Msg 35330, Level 15, State 1, Line 1
INSERT statement failed because data cannot be updated in a table with a columnstore index.

Columnstore over the years

SQL Server 2014 Enterprise Only

Read-only Non-Clustered Columnstore Index

OR

Read/Write Clustered Columnstore Index

Optional Archive Compression

Columnstore over the years

SQL Server 2016 SP1 **Standard Edition**

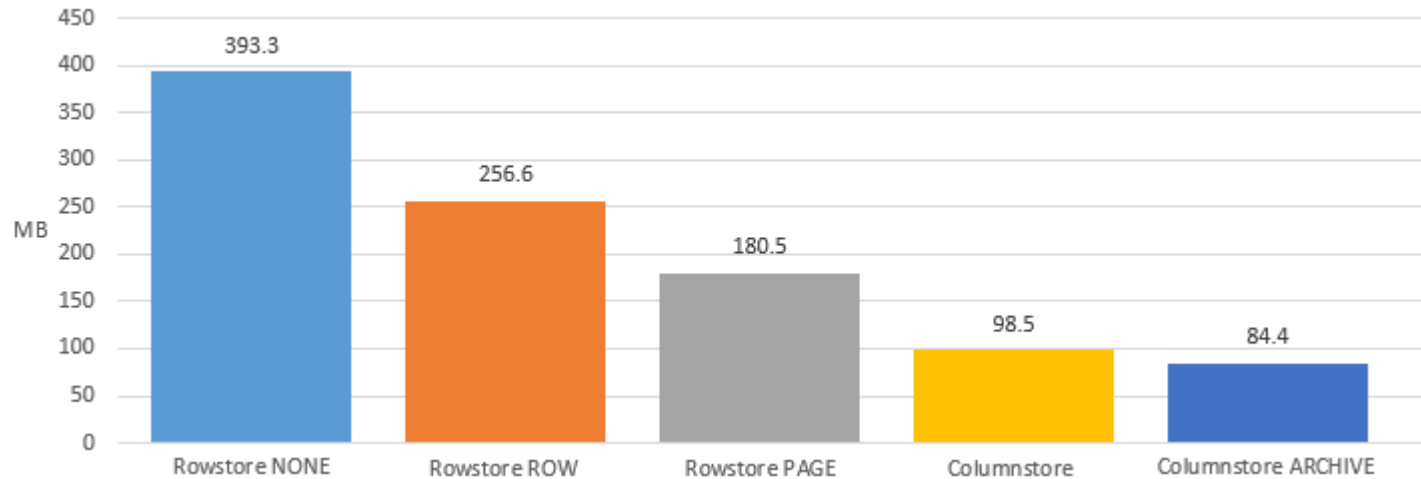
Read/Write Non-Clustered Columnstore Index
OR

Read/Write Clustered Columnstore Index
Now with non-clustered rowstore indexes

Optional Archive Compression

Columnstore Compression

Sample: 4.8 Million row table



We made it!

Thank you for coming!!

If I've done this right we should have time for questions.

Eric Blinn

[linkedin.com/in/ericblinn](https://www.linkedin.com/in/ericblinn)

ericblinn.com

Twitter: [@SQL2TheSequel](https://twitter.com/SQL2TheSequel)