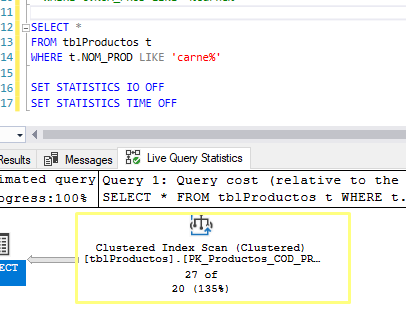
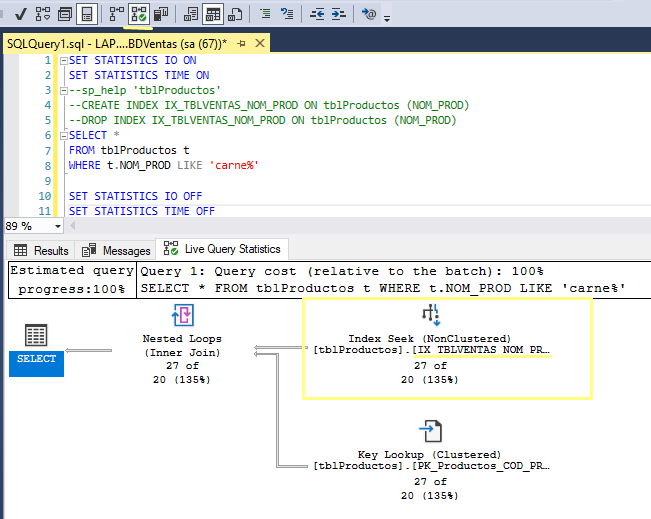
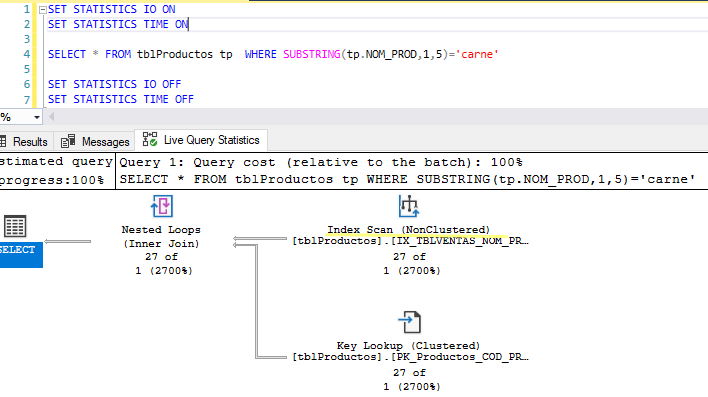
1. Sin Indice cuando hace un barrido de la tabla y NO TIENE INDICE



2) Cuando tiene un Índice bien usadito

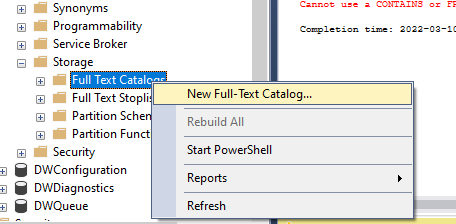
**Index Seek** => si usa el Indice

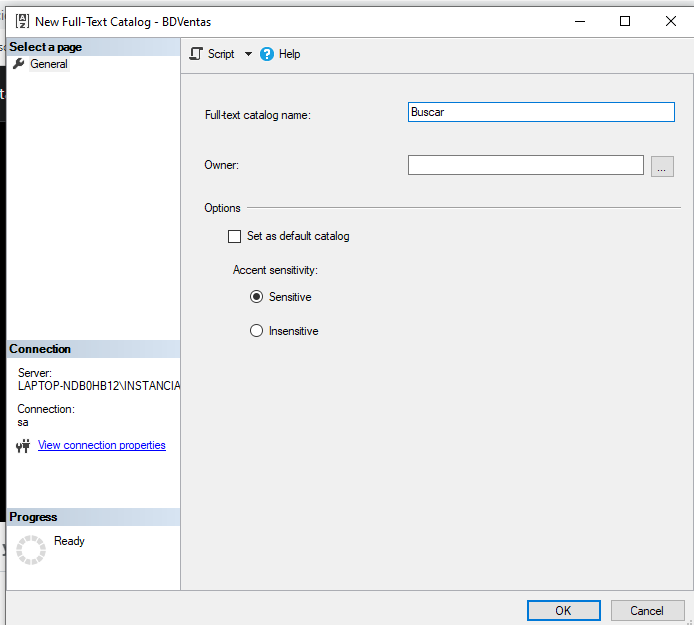
3 Plan de Ejecución,NON-SARGABLE **NO USA INDICE CUANDO HAY FUNCIONES DE CADENA**



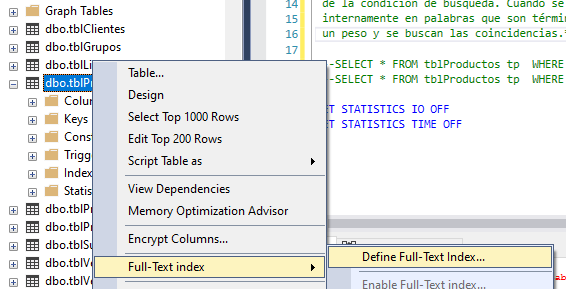
**Index Scan** => No usa el Indice

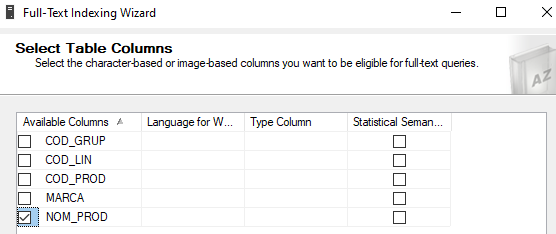
4) **Mejora del Like con full text**

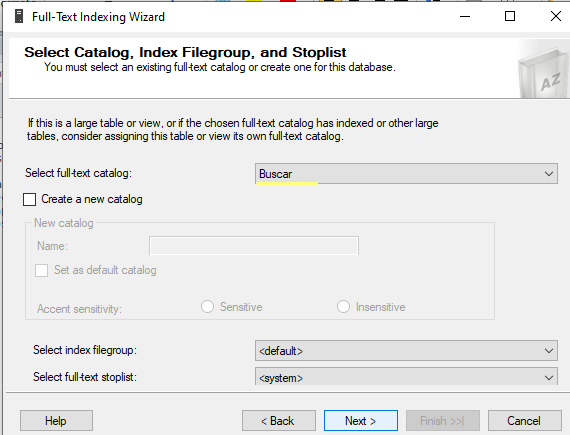


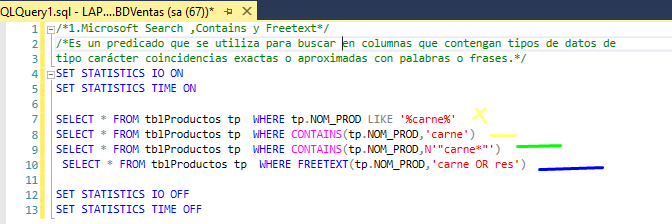


Ir a la tabla







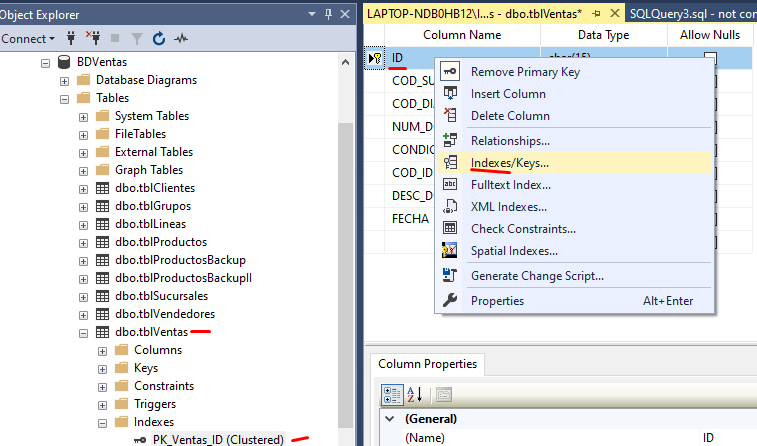


**INDICES DE COBERTURA**

CREATE INDEX IX\_TBLVENTAS\_NOM\_PROD ON tblProductos (NOM\_PROD) INCLUDE(COD\_GRUP,COD\_LIN,MARCA,COS\_PROM\_C,PRECIO\_VTA)

WITH (DROP\_EXISTING=ON)

**INDICE CLUSTER CAMPO FECHA**



CREATE CLUSTERED INDEX IX\_VENTAS\_FECHA ON [dbo].[tblVentas](FECHA)

**FORZAR EL USO DE INDICES**  WITH (FORCESEEK)

* Se crea índice de cobertura

CREATE INDEX IX\_TBLVENTAS\_DETALLE\_ID ON tblVentas\_Detalle(ID) INCLUDE (CANTIDAD,VALOR)

SELECT year(tv.fecha) AS ANNO,

CAST(MONTH(TV.FECHA) AS varchar(10)) + '-'+ DATENAME(mm,tv.FECHA) AS MES,

SUM(tvd.CANTIDAD\*tvd.valor) AS TOTALVENTAS

from tblVentas tv WITH (FORCESEEK) INNER JOIN

tblVentas\_Detalle tvd WITH (FORCESEEK)

ON tv.ID = tvd.ID

WHERE TV.FECHA BETWEEN '01/01/2014' AND '31/12/2015'

GROUP BY year(tv.fecha),MONTH(TV.FECHA),DATENAME(mm,tv.FECHA)

**OPTIMIZACION CON INDECE CLUSTER**

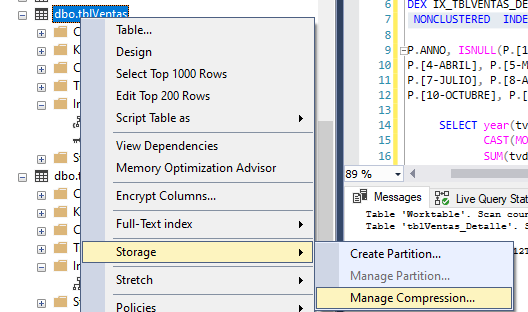
CREATE CLUSTERED INDEX IX\_TBLVENTAS\_DETALLE\_ID ON tblVentas\_Detalle(ID,CANTIDAD,VALOR)

**FILL FACTOR OH FACTOR DE RELLENO**

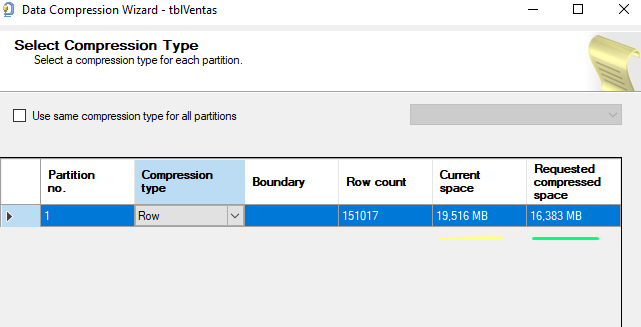
CREATE NONCLUSTERED INDEX IX\_TBLVENTAS\_DETALLE\_ID ON TBLVENTAS\_DETALLE(ID) INCLUDE(CANTIDAD,VALOR) WITH(FILLFACTOR=90)

**COMPRENCION DE DATOS**

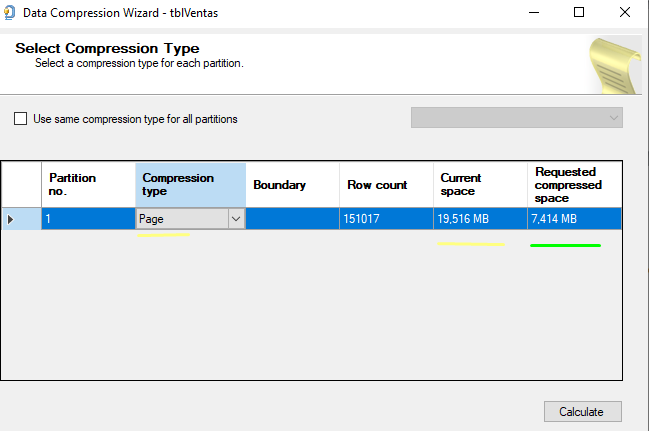
Página=> Unidad mínima de almacenamiento.



A nivel de fila reduce poquito



**A nivel de página reduce casi la mitad mejor opción**



USE [BDVentas]

ALTER TABLE [dbo].[tblVentas] REBUILD PARTITION = ALL

WITH

(DATA\_COMPRESSION = PAGE

)

GO

USE [BDVentas]

ALTER TABLE [dbo].[tblVentas\_dETALLE] REBUILD PARTITION = ALL

WITH

(DATA\_COMPRESSION = PAGE

)

**A NIVEL DE FILA**

CREATE NONCLUSTERED INDEX IX\_TBLVENTAS\_DETALLE\_ID ON TBLVENTAS\_DETALLE(ID) INCLUDE(CANTIDAD,VALOR) WITH(DATA\_COMPRESSION=ROW)

Esta opción baja las lecturas lógicas de 15000 a 12000

**A NIVEL DE PAGINA (Mejor Opcion)**

CREATE NONCLUSTERED INDEX IX\_TBLVENTAS\_DETALLE\_ID ON TBLVENTAS\_DETALLE(ID) INCLUDE(CANTIDAD,VALOR) WITH(DATA\_COMPRESSION=PAGE)

Esta opción baja las lecturas lógicas de 15000 a 9200

**INDICES REDUNDANTES**

Si se piden más campos en la consulta es mejor adicionarlos en un solo índice en vez de crear más.

CREATE NONCLUSTERED INDEX IX\_TBLVENTAS\_DETALLE\_ID ON TBLVENTAS\_DETALLE(ID) INCLUDE(CANTIDAD,VALOR,COSTO) WITH(DATA\_COMPRESSION=PAGE)

**INDICES COLUMNARES** (Recomendados para Datawarerhouse BI tabla de hechos y cuando tiene millones de registros)

CREATE CLUSTERED COLUMNSTORE INDEX IX\_TBLVENTAS\_DETALLE ON tblVentas\_Detalle

Se evidencia que ya no hay lecturas lógicas en los **índices** **columnstore**

\*\*\*\*\* Solo se puede en versiones **Enterprise y Developer**.

**PARTE 2**

DROP INDEX IX\_tblVentas\_Detalle ON tblVentas\_Detalle

CREATE COLUMNSTORE INDEX IX\_tblVentas\_Detalle ON tblVentas\_Detalle(

[ID],[COD\_PROD],[NUM\_LIN],[CANTIDAD],[COSTO],[VALOR])

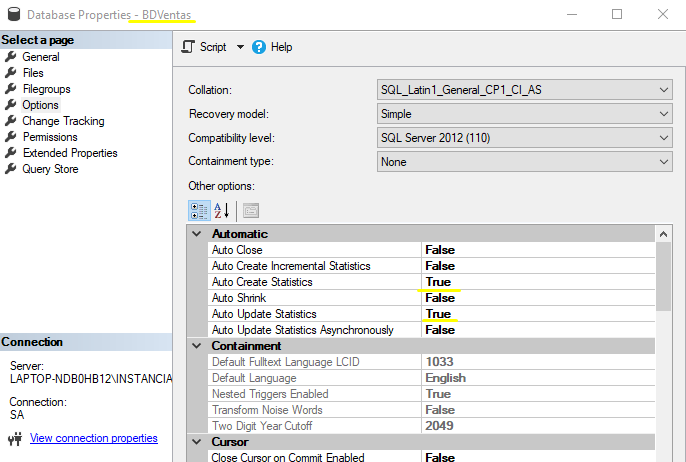
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* OPTIMIZAR EL SCRIPT 17 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

AVERIGUAR PLAN DE EJECUCION COLLATE DATABASE\_DEFAULT

Preocuparse por table scan hash join

**TUNNING ADVISOR**

**ESTADISTICAS SQLSERVER**



DBCC SHOW\_STATISTICS ('tblproductos','IX\_TBLVENTAS\_NOM\_PROD') with stat\_header

DBCC SHOW\_STATISTICS ('tblproductos','IX\_TBLVENTAS\_NOM\_PROD') with density\_vector

update STATISTICS tblproductos with fullscan

**PARALELISMO**

OPTION (MAXDOP 1) =🡺 Al Final

SET STATISTICS IO ON

SET STATISTICS TIME ON

SET DATEFORMAT DMY;

SELECT ts.Sucursal,

tv.COD\_DIA,

tv.NUM\_DOC,

tv.CONDICIONES,

tz.ZONA,

tv2.NOM\_VEND AS VENDEDOR,

tc.NOMBRE AS CLIENTE,

tp.NOM\_PROD,

tl.NOM\_LIN AS LINEA,

tg.NOM\_GRUP AS GRUPO,

tvd.CANTIDAD\* tvd.VALOR AS TOTAL\_VENTA,

tvd.CANTIDAD\* tvd.COSTO AS TOTAL\_COSTO,

tvd.CANTIDAD\* tvd.VALOR- tvd.CANTIDAD\* tvd.COSTO AS TOTAL\_UTILIDAD

FROM tblVentas tv INNER JOIN tblVentas\_Detalle tvd ON tv.ID = tvd.ID

INNER JOIN tblSucursales ts ON tv.COD\_SUC = ts.COD\_SUC COLLATE DATABASE\_DEFAULT

INNER JOIN tblClientes tc ON tv.COD\_ID = tc.COD\_ID COLLATE DATABASE\_DEFAULT

INNER JOIN tblProductos tp ON tvd.COD\_PROD = tp.COD\_PROD COLLATE DATABASE\_DEFAULT

INNER JOIN tblLineas tl ON tp.COD\_LIN = tl.COD\_LIN

INNER JOIN tblGrupos tg ON tp.COD\_GRUP = tg.COD\_GRUP

INNER JOIN tblVendedores tv2 ON tc.COD\_VEND = tv2.COD\_VEND

INNER JOIN tblZonas tz ON tc.COD\_ZON = tz.COD\_ZON

WHERE TV.FECHA BETWEEN '01/01/2014' AND '31/12/2015'

OPTION (MAXDOP 1)

SET STATISTICS IO OFF

SET STATISTICS TIME OFF

**FORZAR OPERACIÓN LOOP JOIN, MERGE JOIN Y HASH JOIN**

Hay que probar las 3 operaciones haber cual funciona mejor **LOOP, MERGE Y HASH**

OPTION (LOOP JOIN)

/\*1.Grado de Paralelismo\*/

SET STATISTICS IO ON

SET STATISTICS TIME ON

SET DATEFORMAT DMY;

SELECT ts.Sucursal,

tv.COD\_DIA,

tv.NUM\_DOC,

tv.CONDICIONES,

tz.ZONA,

tv2.NOM\_VEND AS VENDEDOR,

tc.NOMBRE AS CLIENTE,

tp.NOM\_PROD,

tl.NOM\_LIN AS LINEA,

tg.NOM\_GRUP AS GRUPO,

tvd.CANTIDAD\* tvd.VALOR AS TOTAL\_VENTA,

tvd.CANTIDAD\* tvd.COSTO AS TOTAL\_COSTO,

tvd.CANTIDAD\* tvd.VALOR- tvd.CANTIDAD\* tvd.COSTO AS TOTAL\_UTILIDAD

FROM tblVentas tv INNER JOIN tblVentas\_Detalle tvd ON tv.ID = tvd.ID

INNER JOIN tblSucursales ts ON tv.COD\_SUC = ts.COD\_SUC COLLATE DATABASE\_DEFAULT

INNER JOIN tblClientes tc ON tv.COD\_ID = tc.COD\_ID COLLATE DATABASE\_DEFAULT

INNER JOIN tblProductos tp ON tvd.COD\_PROD = tp.COD\_PROD COLLATE DATABASE\_DEFAULT

INNER JOIN tblLineas tl ON tp.COD\_LIN = tl.COD\_LIN

INNER JOIN tblGrupos tg ON tp.COD\_GRUP = tg.COD\_GRUP

INNER JOIN tblVendedores tv2 ON tc.COD\_VEND = tv2.COD\_VEND

INNER JOIN tblZonas tz ON tc.COD\_ZON = tz.COD\_ZON

WHERE TV.FECHA BETWEEN '01/01/2014' AND '31/12/2015'

OPTION (LOOP JOIN)

SET STATISTICS IO OFF

SET STATISTICS TIME OFF

==================================================================================

OPTION (MERGE JOIN)

==================================================================================

/\*1.Grado de Paralelismo\*/

SET STATISTICS IO ON

SET STATISTICS TIME ON

SET DATEFORMAT DMY;

SELECT ts.Sucursal,

tv.COD\_DIA,

tv.NUM\_DOC,

tv.CONDICIONES,

tz.ZONA,

tv2.NOM\_VEND AS VENDEDOR,

tc.NOMBRE AS CLIENTE,

tp.NOM\_PROD,

tl.NOM\_LIN AS LINEA,

tg.NOM\_GRUP AS GRUPO,

tvd.CANTIDAD\* tvd.VALOR AS TOTAL\_VENTA,

tvd.CANTIDAD\* tvd.COSTO AS TOTAL\_COSTO,

tvd.CANTIDAD\* tvd.VALOR- tvd.CANTIDAD\* tvd.COSTO AS TOTAL\_UTILIDAD

FROM tblVentas tv INNER JOIN tblVentas\_Detalle tvd ON tv.ID = tvd.ID

INNER JOIN tblSucursales ts ON tv.COD\_SUC = ts.COD\_SUC COLLATE DATABASE\_DEFAULT

INNER JOIN tblClientes tc ON tv.COD\_ID = tc.COD\_ID COLLATE DATABASE\_DEFAULT

INNER JOIN tblProductos tp ON tvd.COD\_PROD = tp.COD\_PROD COLLATE DATABASE\_DEFAULT

INNER JOIN tblLineas tl ON tp.COD\_LIN = tl.COD\_LIN

INNER JOIN tblGrupos tg ON tp.COD\_GRUP = tg.COD\_GRUP

INNER JOIN tblVendedores tv2 ON tc.COD\_VEND = tv2.COD\_VEND

INNER JOIN tblZonas tz ON tc.COD\_ZON = tz.COD\_ZON

WHERE TV.FECHA BETWEEN '01/01/2014' AND '31/12/2015'

OPTION (MERGE JOIN)

SET STATISTICS IO OFF

SET STATISTICS TIME OFF

==================================================================================

OPTION (HASH JOIN)

==================================================================================

/\*1.Grado de Paralelismo\*/

SET STATISTICS IO ON

SET STATISTICS TIME ON

SET DATEFORMAT DMY;

SELECT ts.Sucursal,

tv.COD\_DIA,

tv.NUM\_DOC,

tv.CONDICIONES,

tz.ZONA,

tv2.NOM\_VEND AS VENDEDOR,

tc.NOMBRE AS CLIENTE,

tp.NOM\_PROD,

tl.NOM\_LIN AS LINEA,

tg.NOM\_GRUP AS GRUPO,

tvd.CANTIDAD\* tvd.VALOR AS TOTAL\_VENTA,

tvd.CANTIDAD\* tvd.COSTO AS TOTAL\_COSTO,

tvd.CANTIDAD\* tvd.VALOR- tvd.CANTIDAD\* tvd.COSTO AS TOTAL\_UTILIDAD

FROM tblVentas tv INNER JOIN tblVentas\_Detalle tvd ON tv.ID = tvd.ID

INNER JOIN tblSucursales ts ON tv.COD\_SUC = ts.COD\_SUC COLLATE DATABASE\_DEFAULT

INNER JOIN tblClientes tc ON tv.COD\_ID = tc.COD\_ID COLLATE DATABASE\_DEFAULT

INNER JOIN tblProductos tp ON tvd.COD\_PROD = tp.COD\_PROD COLLATE DATABASE\_DEFAULT

INNER JOIN tblLineas tl ON tp.COD\_LIN = tl.COD\_LIN

INNER JOIN tblGrupos tg ON tp.COD\_GRUP = tg.COD\_GRUP

INNER JOIN tblVendedores tv2 ON tc.COD\_VEND = tv2.COD\_VEND

INNER JOIN tblZonas tz ON tc.COD\_ZON = tz.COD\_ZON

WHERE TV.FECHA BETWEEN '01/01/2014' AND '31/12/2015'

OPTION (HASH JOIN)

SET STATISTICS IO OFF

SET STATISTICS TIME OFF

**MANTENIMIENTO DE INDICES**

Hay