

Informe Performance Base de datos Sucursal

Agosto 2024



- Configuración de parámetros.
- Estadística inicial y final del parámetro: Cost threshold for parallelism.
- Estadística inicial y final del parámetro: min memory per query (KB).
- Estadística inicial y final del parámetro: AUTO_SHRINK (fragmentación).
- Estadística inicial y final del parámetro: FILEGROWTH.
- Conclusiones

Configuración de Parámetros

La muestra de los resultados de las pruebas previas y posteriores a los cambios en configuración de la base de datos del store Farmax_local se obtuvieron realizando transacciones y consultas hacia las siguientes tablas.

El proceso de transaccional insert/update hacia las tablas se hizo replicando el proceso de venta mostrador y SAD con pago en efectivo de forma automática en 50,000 ocasiones de forma recurrente, sin uso del monedero electrónico.

El proceso de consultas hacia las tablas se enfocó a la búsqueda de productos en todas sus variantes, por nombre, de acuerdo con la sal, etc.

Se omitieron procesos de servicios a terceros como lo es cobro con tarjeta (débito o crédito), pago de servicio a terceros, consulta de programa de lealtad, cobro con monedero, ya que no forman parte de la base de datos del store.

No se realiza el proceso de cierre del día ya que las tablas involucradas no cuentan con una columna que identifique que el registro es prueba, por ejemplo, estaActivo o estaModoEntrenamiento y esto puede afectar alguna sucursal de laboratorio.

Parámetro	Valor inicial	Valor sugerido	Resultado esperado
Cost threshold for parallelism	5	50	Mejorar el costo de cada plan de ejecución por consulta.
Min memory per query (KB)	1024 KB	2048 KB	Evitar consumo alto de lectura y escritura de páginas de base de datos.
Min server memory (MB)	0 MB	256 MB	Evitar consumo alto de memoria física y lógica al momento de utilizar índices tipo cluster y no clustered al insertar y/o actualizar información.
AUTO_SHRINK	OFF	ON	Evitar que SQL Server realice de forma automática la fragmentación de páginas al realizar una inserción y/o actualización de datos.
FILEGROWTH	1 MB	100 MB	Que el crecimiento de la base de datos no sea una limitante al momento de realizar de forma paralela una gran cantidad de transacciones en la sucursal.

Uso de memoria física de SQL Server

Configuración de parámetro: min server memory (MB)

La muestra de los resultados se obtuvo afectando las siguientes tablas realizando inserciones y/o actualización de registros, además de consultas comunes que realizan en una sucursal sin conexión hacia el servidor central y/o servicios de externos (programa de lealtad, monedero, etc.).

Tablas afectadas	
venFechaOperacion	venDevolucionesCab
venTurnosCajaReg	venDevolucionesDet
venCortesCab	genDonativosEspeciales
venCortesDet	sadOrdenSeguimiento
genMovimientosCajasCab	sadRegistroLiquidacion
genMovimientosCajasDet	
trnTransaccionesCab	
trnTransaccionesDet	

Uso de memoria física de SQL Server

Configuración de parámetro: min server memory (MB)

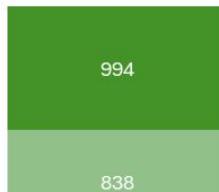
En cada muestra de pruebas se realizaron las siguientes acciones

1. Reinicio de servicio de Windows MSSQLSERVER.
2. Se realizan 50,000 inserciones en las tablas afectadas.
3. Toma de valores de memoria física y virtual posteriores a la inserción de registros.
4. Ajuste de configuración "min server memory (MB)" a 256 MB.
5. Reinicio de servicio de Windows MSSQLSERVER.
6. Se realizan 50,000 inserciones en las tablas afectadas.
7. Toma de valores de memoria física y virtual posteriores a la inserción de registros.

Valor Inicial: 0 MB

ESTADISTICA INICIAL

■ Antes de transacciones ■ Despues de transacciones



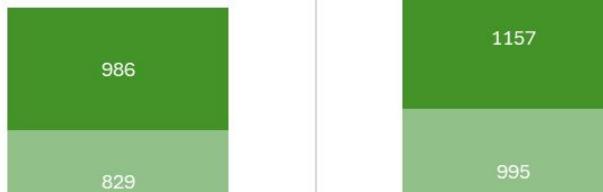
MEMORIA FISICA (MB)

MEMORIA VIRTUAL (MB)

Valor sugerido: 256 MB

ESTADISTICA FINAL

■ Antes de transacciones ■ Despues de transacciones



MEMORIA FISICA (MB)

MEMORIA VIRTUAL (MB)

Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Las muestras de resultados se realizaron en base a un grupo de consultas que se realizan en la sucursal.

Se identificaron diversas consultas en procesos base como lo son

1. Promociones por sucursal.
2. Configuraciones asignadas a sucursal.
3. Identificar movimientos como ingresos, devoluciones, donativos especiales para realizar el cierre del día.

Tablas afectadas		
genSucursalesCat	genPromocionesCupon	genMovimientosCajasMsiDet
genAgrupacionesSucursalesCat	cnfConfiguracionAgrupacionSucursalesCnf	cnfParametrosConfiguracionAplicacion
genAsociacionesAgrupacionSucursales	genConfiguracionAgrupacionSucursalesReg	genDonativosEspeciales
genAgrupacionSucursalesPromociones	trnTransaccionesCab	venDevolucionesCab
genPromociones	rtlPosFeFinalOr	sadOrdenSeguimiento
genReglasPrecioPromocion	genMovimientosCajasCab	sadRegistroLiquidacion
genPeriodosGruposPeriodo	trnTransaccionesDet	
genPeriodos	genMovimientosCajasDet	

Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

En cada muestra de pruebas se realizaron las siguientes acciones

1. Validar y confirmar que el valor de la propiedad "cost threshold for parallelism" de la base de datos se encuentre su límite en 5.
2. Ciclo #1, se realizan prueba de consultas detectadas.
3. Toma de valores de uso de cantidades mínimas y máximas de: lecturas físicas y lógicas de páginas, escrituras lógicas que realiza el plan de ejecución.
4. Se realiza configuración de la propiedad "cost threshold for parallelism" de la base de datos se encuentre con un límite de 50.
5. Reinicio de servicio de Windows MSSQLSERVER.
6. Ciclo #2, se realizan prueba de consultas detectadas.
7. Toma de valores de uso de cantidades mínimas y máximas de: lecturas físicas y lógicas de páginas, escrituras lógicas que realiza el plan de ejecución.

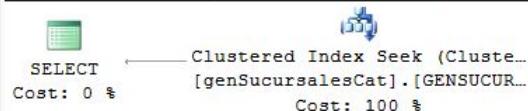
Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 5: Promociones por sucursal

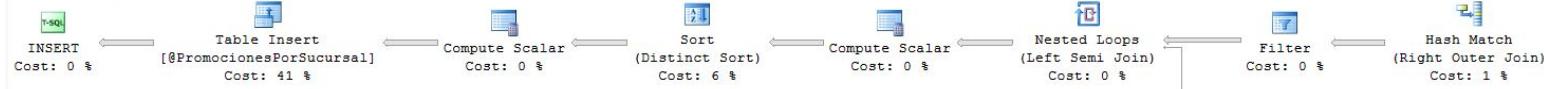
Query 6: Query cost (relative to the batch): 0%

```
SELECT @esFranquicia = esFranquicia FROM genSucursalesCat AS gsc WHERE gsc.codigoSucursal = @idSucursal
```



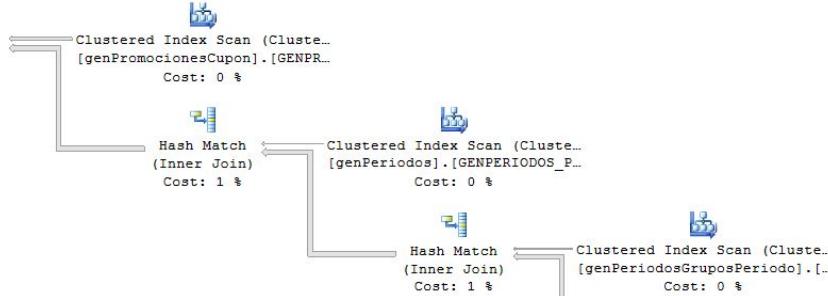
Query 7: Query cost (relative to the batch): 100%

```
;WITH CTE_Agrupaciones(idAgrupacion) AS ( SELECT casr.codigoAgrupacion AS idAgrupacion FROM genAgrupacionesSucursalesCat AS casr INNER JOIN q...
```



Query 7: Query cost (relative to the batch): 100%

```
;WITH CTE_Agrupaciones(idAgrupacion) AS ( SELECT casr.codigoAgrupacion AS idA...
```

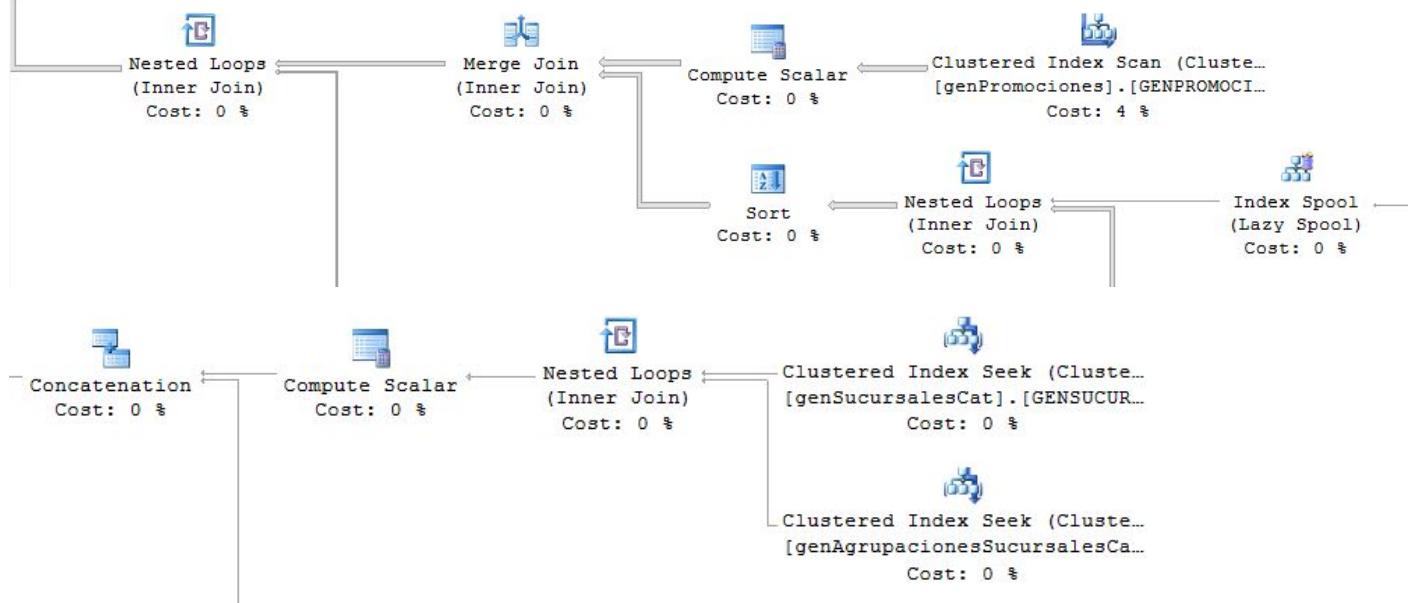


Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 5: Promociones por sucursal

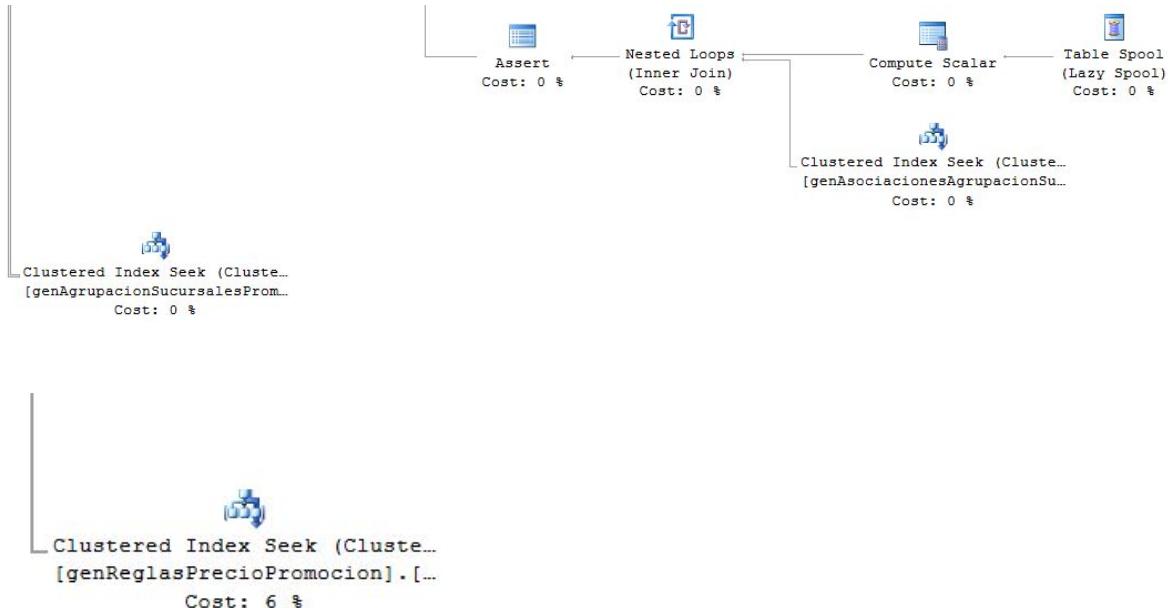
Query 7: Query cost (relative to the batch): 100%
;WITH CTE_Agrupaciones(idAgrupacion) AS (SELECT casr.codigoAgrupacion AS idAgrupacion FROM genAg



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 5: Promociones por sucursal

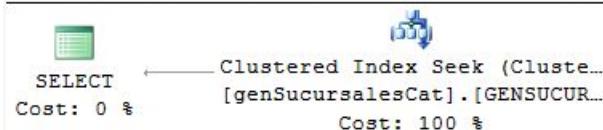


Limitar el costo de paralelismo SQL Server

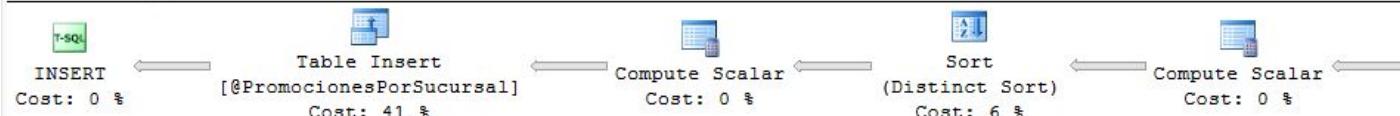
Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 50: Promociones por sucursal

Query 6: Query cost (relative to the batch): 0%
SELECT @esFranquicia = esFranquicia FROM genSucursalesCat



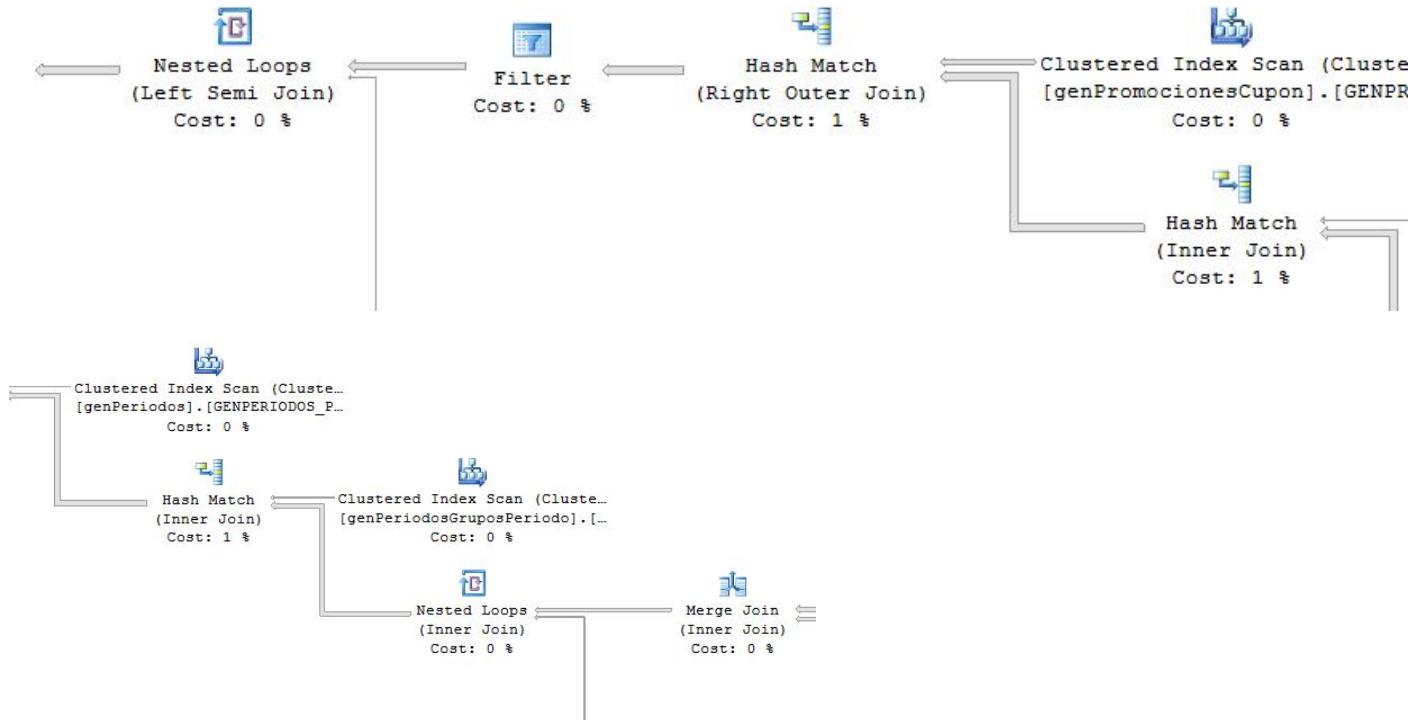
Query 7: Query cost (relative to the batch): 100%
;WITH CTE_Agrupaciones(idAgrupacion) AS (SELECT casr.codigoAgrupacion AS idAgrupacion FROM casr)
INSERT INTO @PromocionesPorSucursal
SELECT * FROM CTE_Agrupaciones
INNER JOIN promociones
ON CTE_Agrupaciones.idAgrupacion = promociones.idAgrupacion
WHERE sucursal_id = 1
GROUP BY sucursal_id, idAgrupacion, agrupacion_nombre



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

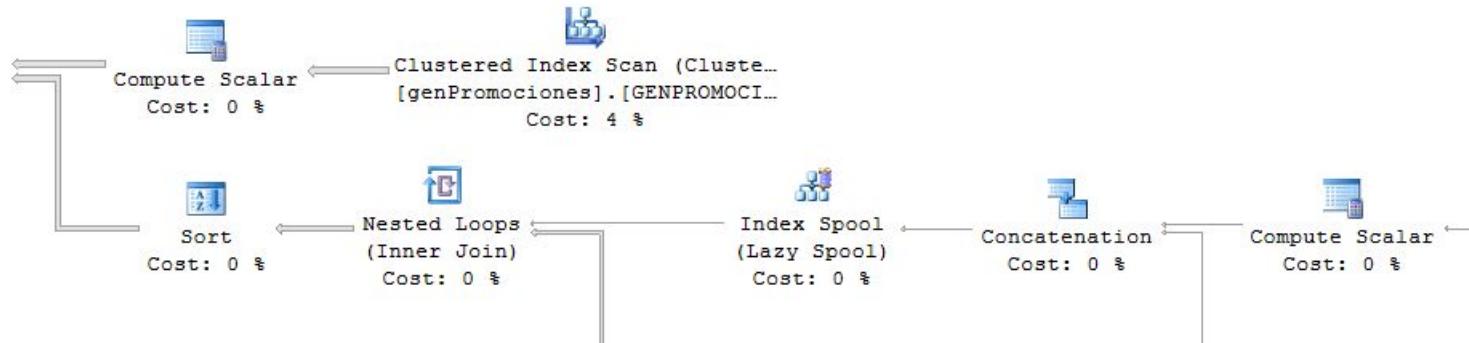
Resultados de las pruebas con límite de paralelismo de 50: Promociones por sucursal



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

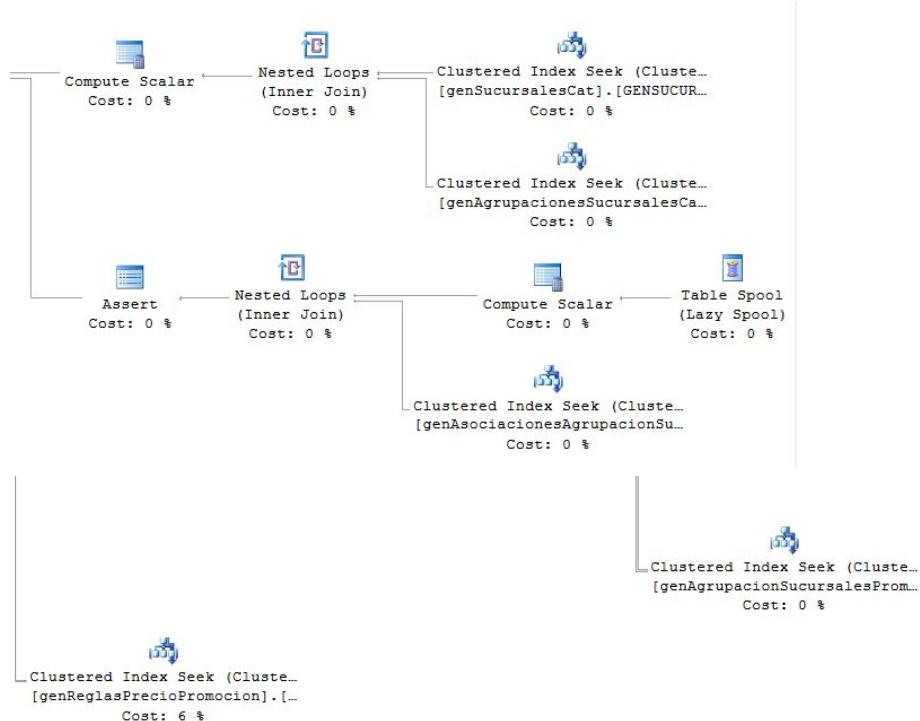
Resultados de las pruebas con límite de paralelismo de 50: Promociones por sucursal



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

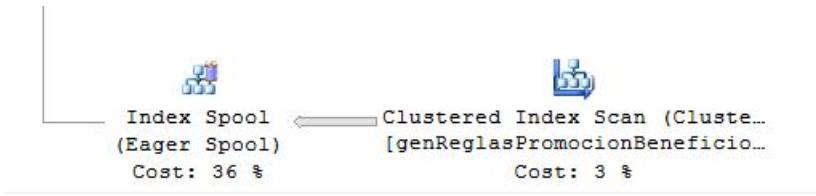
Resultados de las pruebas con límite de paralelismo de 50: Promociones por sucursal



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 50: Promociones por sucursal

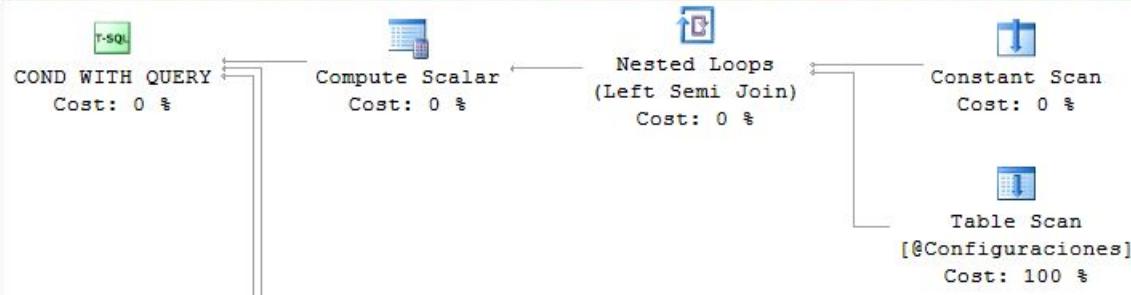


Limitar el costo de paralelismo SQL Server

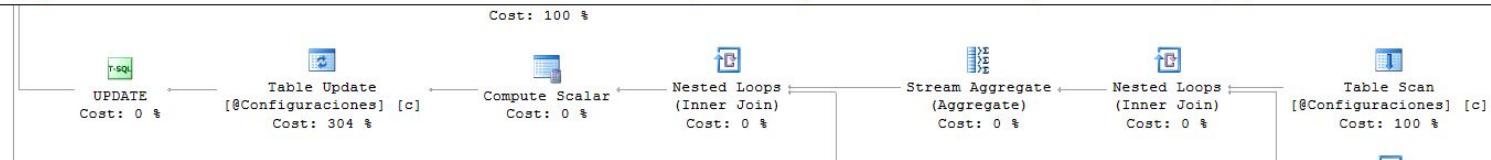
Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 5: Configuraciones por sucursal

```
Query 8: Query cost (relative to the batch): 3%
WHILE (EXISTS (SELECT 1 FROM @Configuraciones WHERE valor IS NULL) AND @idAgrupacion IS NOT NULL)
Missing Index (Impact 91.9482): CREATE NONCLUSTERED INDEX [<Name of Missing Index, sysname,>] ON
```



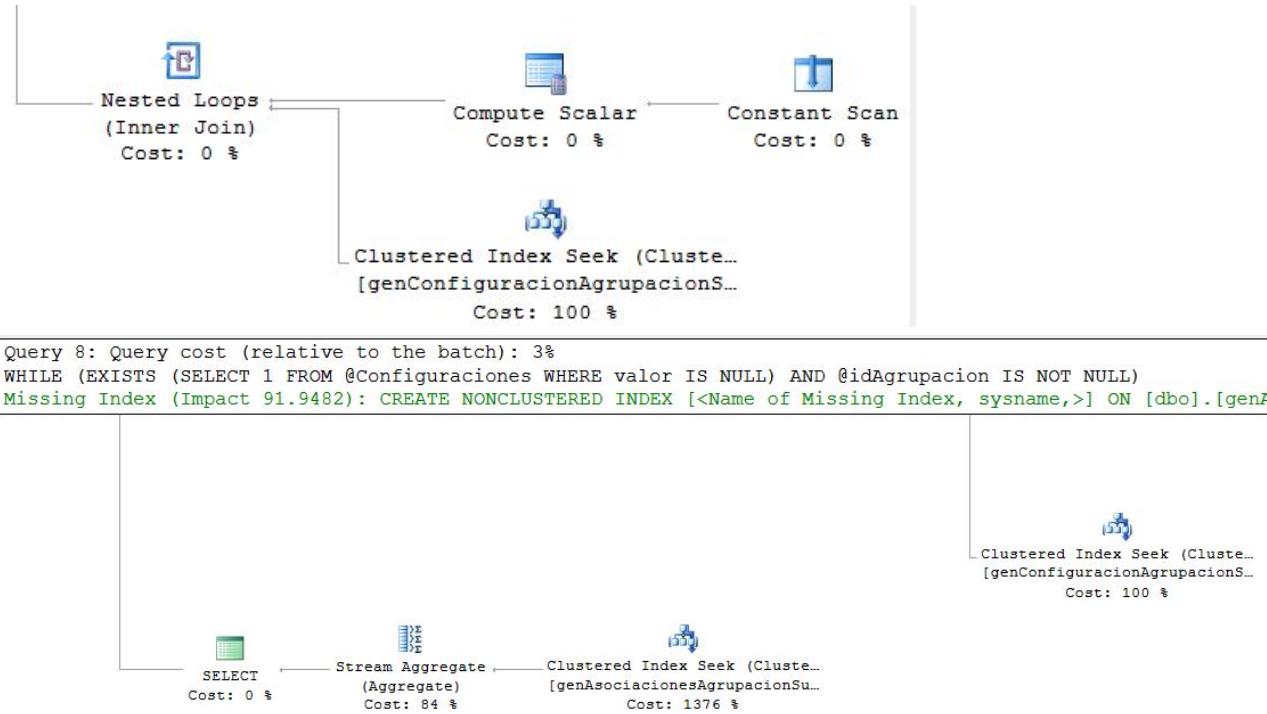
```
Query 8: Query cost (relative to the batch): 3%
WHILE (EXISTS (SELECT 1 FROM @Configuraciones WHERE valor IS NULL) AND @idAgrupacion IS NOT NULL)
Missing Index (Impact 91.9482): CREATE NONCLUSTERED INDEX [<Name of Missing Index, sysname,>] ON [dbo].[genAsociacionesAgrupacionSucursales] ([codig
```



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 5: Configuraciones por sucursal



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 50: Configuraciones por sucursal

Query 5: Query cost (relative to the batch): 3%

```
SELECT @idAgrupacion = idAgrupacion FROM genSucursa  
Missing Index (Impact 91.9482): CREATE NONCLUSTERED
```

```
SELECT [genSucursalesCat].[GENSUCUR...  
Cost: 0 % Cost: 100 %
```

Query 6: Query cost (relative to the batch): 47%

```
DECLARE @Configuraciones AS TABLE ( codigoConfiguracion smallint NOT NULL, parametro NVARCHAR(60) :  
Missing Index (Impact 91.9482): CREATE NONCLUSTERED INDEX [<Name of Missing Index, sysname,>] ON [
```

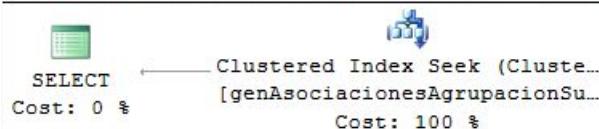
```
INSERT [cnfConfiguracionAgrupacionS...  
Cost: 0 % Cost: 48 % Cost: 0 % Cost: 14 % Cost: 23 % Cost: 15 %
```

Limitar el costo de paralelismo SQL Server

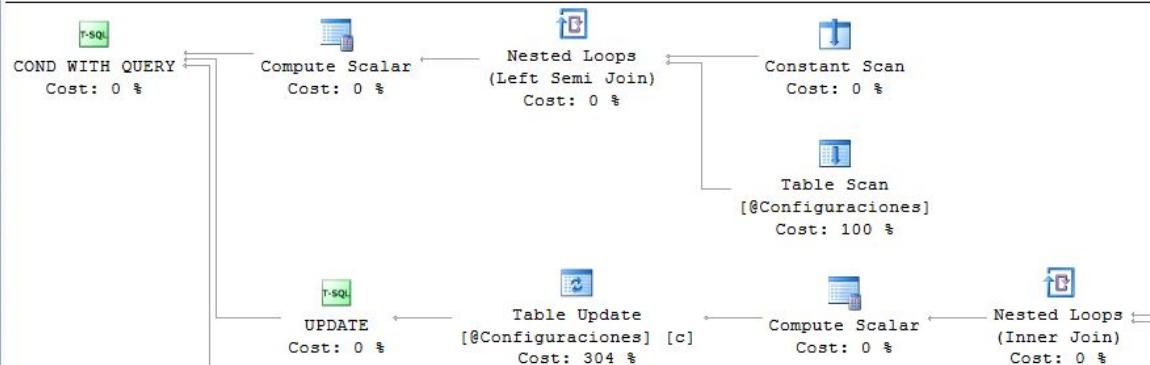
Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 50: Configuraciones por sucursal

Query 7: Query cost (relative to the batch): 44%
SELECT @idAgrupacion=codigoAgrupacionPadre FROM gen
Missing Index (Impact 91.9482): CREATE NONCLUSTERED



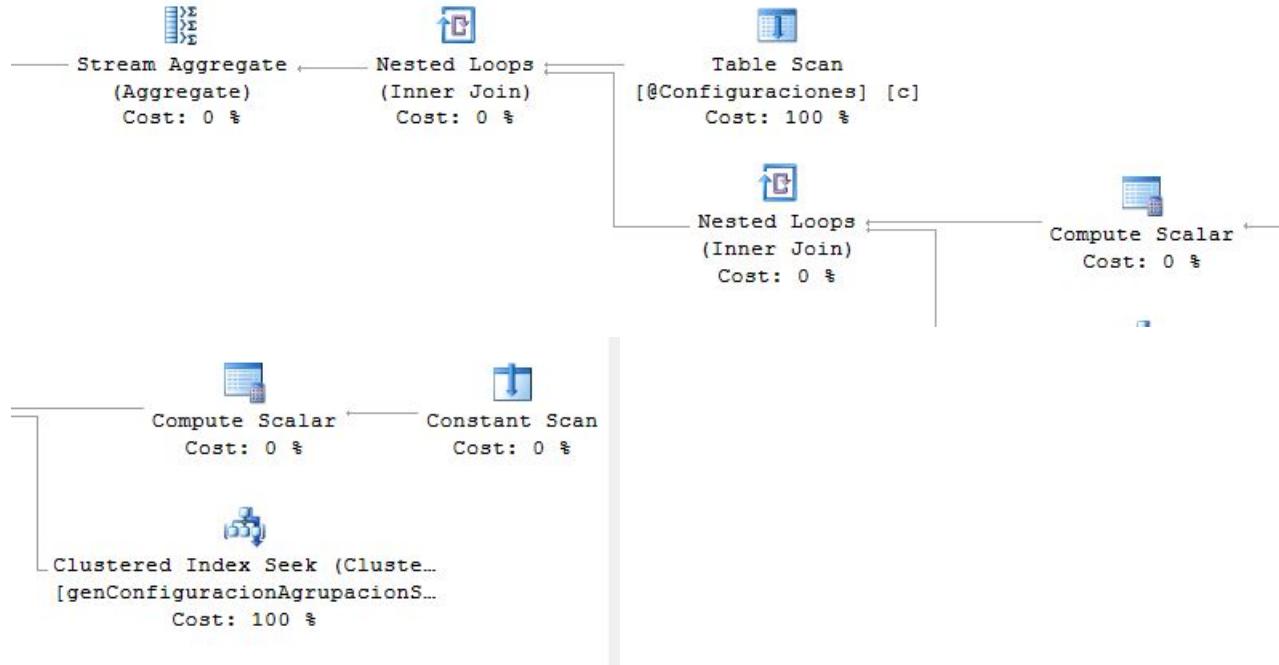
Query 8: Query cost (relative to the batch): 3%
WHILE (EXISTS (SELECT 1 FROM @Configuraciones WHERE valor IS NULL) AND @idAgrupacion :
Missing Index (Impact 91.9482): CREATE NONCLUSTERED INDEX [<Name of Missing Index, sy]



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

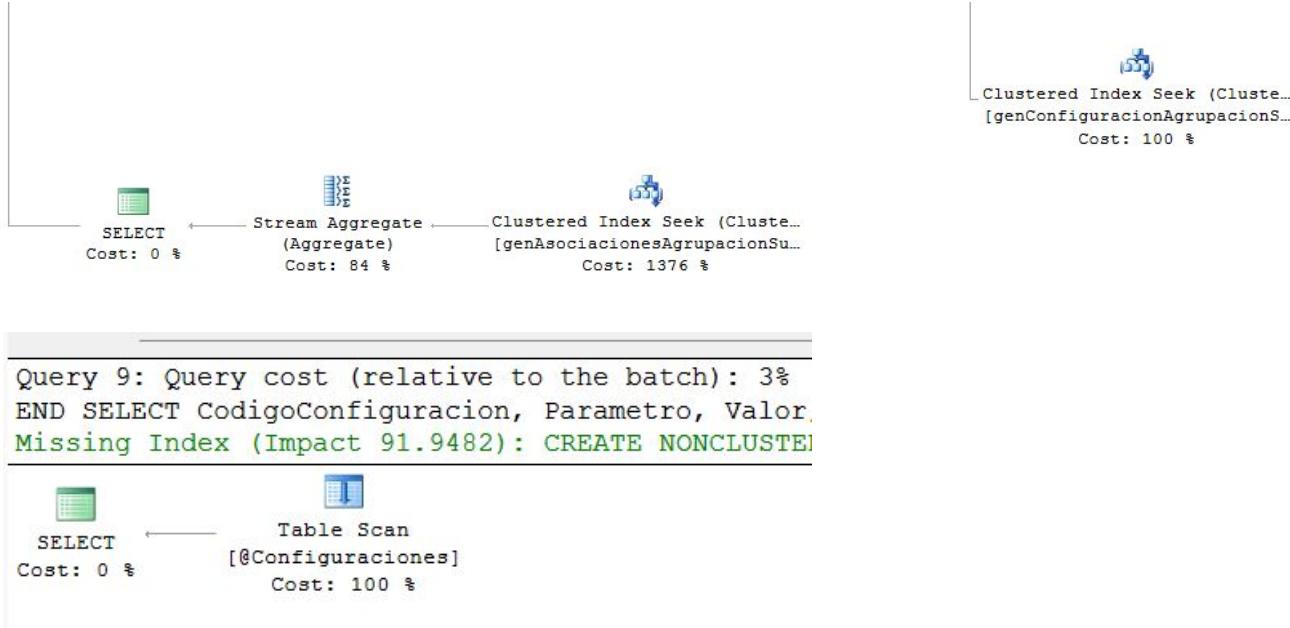
Resultados de las pruebas con límite de paralelismo de 50: Configuraciones por sucursal



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 50: Configuraciones por sucursal

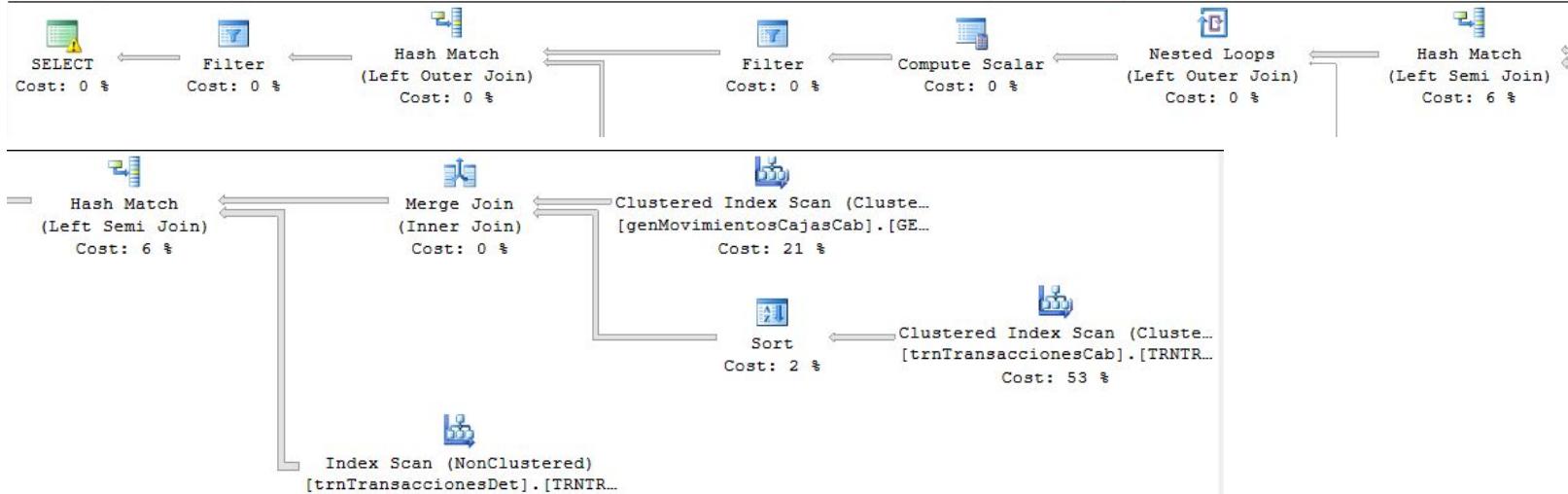


Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 5: Transacciones aplicadas

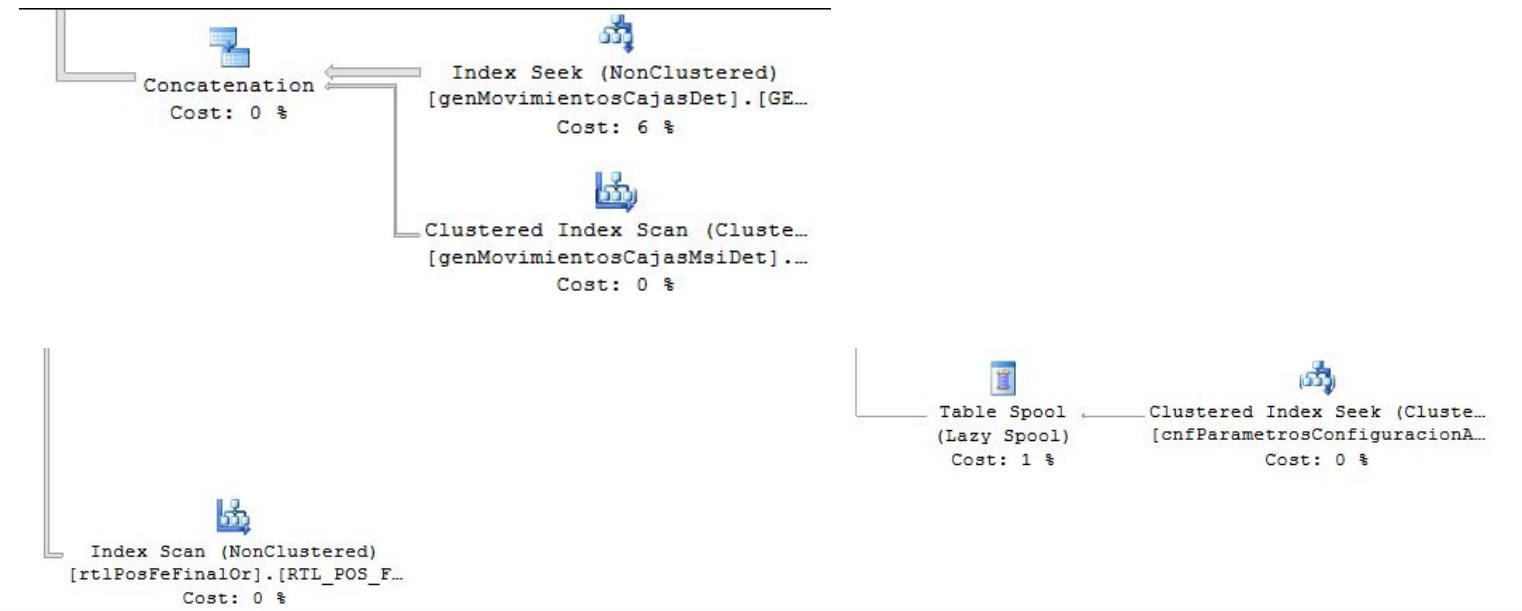
```
Query 3: Query cost (relative to the batch): 100%
SELECT a.idTransaccion FROM trnTransaccionesCab a LEFT JOIN rtlPosFeFinalOr c ON c.movimientoId = a.idTransaccion AND c
Missing Index (Impact 52.4161): CREATE NONCLUSTERED INDEX [<Name of Missing Index, sysname,>] ON [dbo].[trnTransaccionesCab]
```



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 5: Transacciones aplicadas

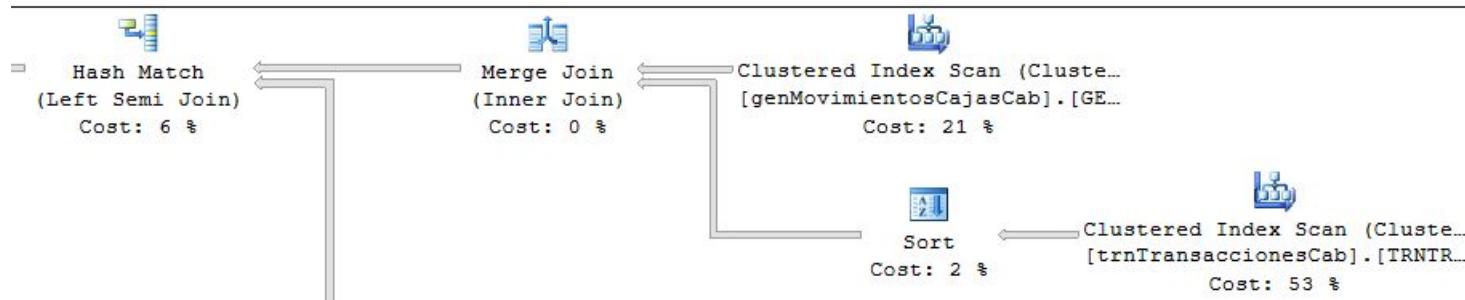


Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 50: Transacciones aplicadas

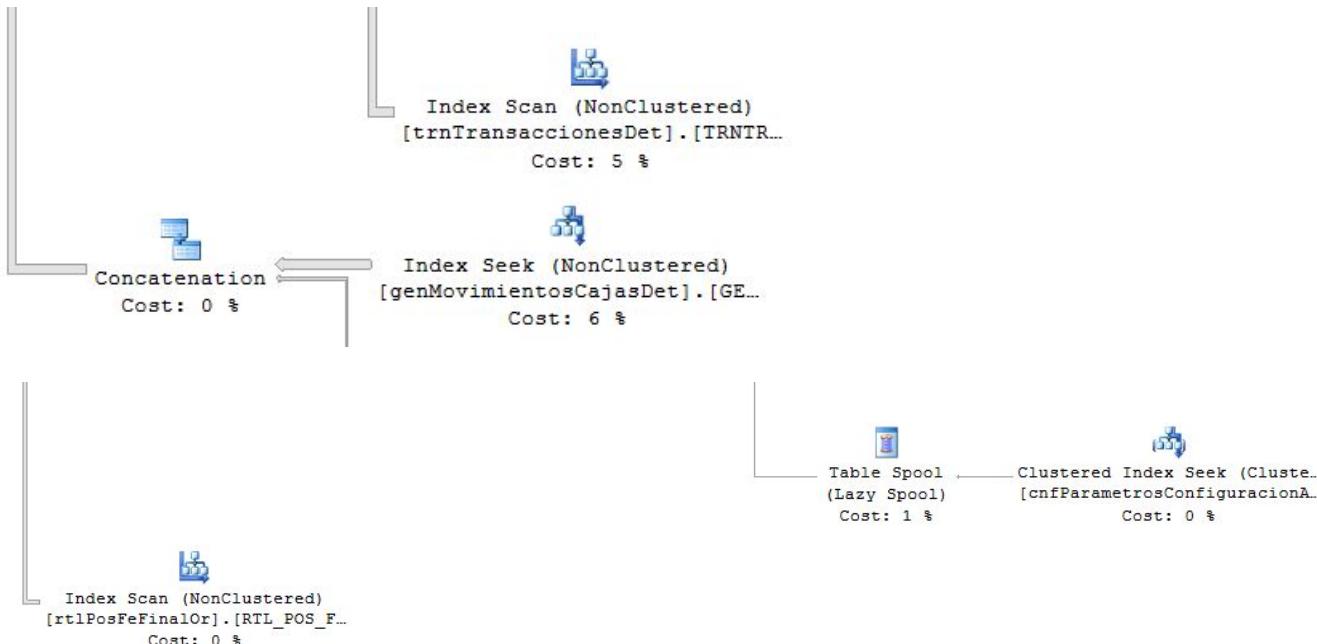
```
Query 3: Query cost (relative to the batch): 100%
SELECT a.idTransaccion FROM trnTransaccionesCab a LEFT JOIN rtlPosFeFinalOr c ON c.movimientoId = a.idTransaccion AND c
Missing Index (Impact 52.4161): CREATE NONCLUSTERED INDEX [<Name of Missing Index, sysname,>] ON [dbo].[trnTransaccionesCab]
    (idTransaccion) INCLUDE (idTransaccion)
```



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

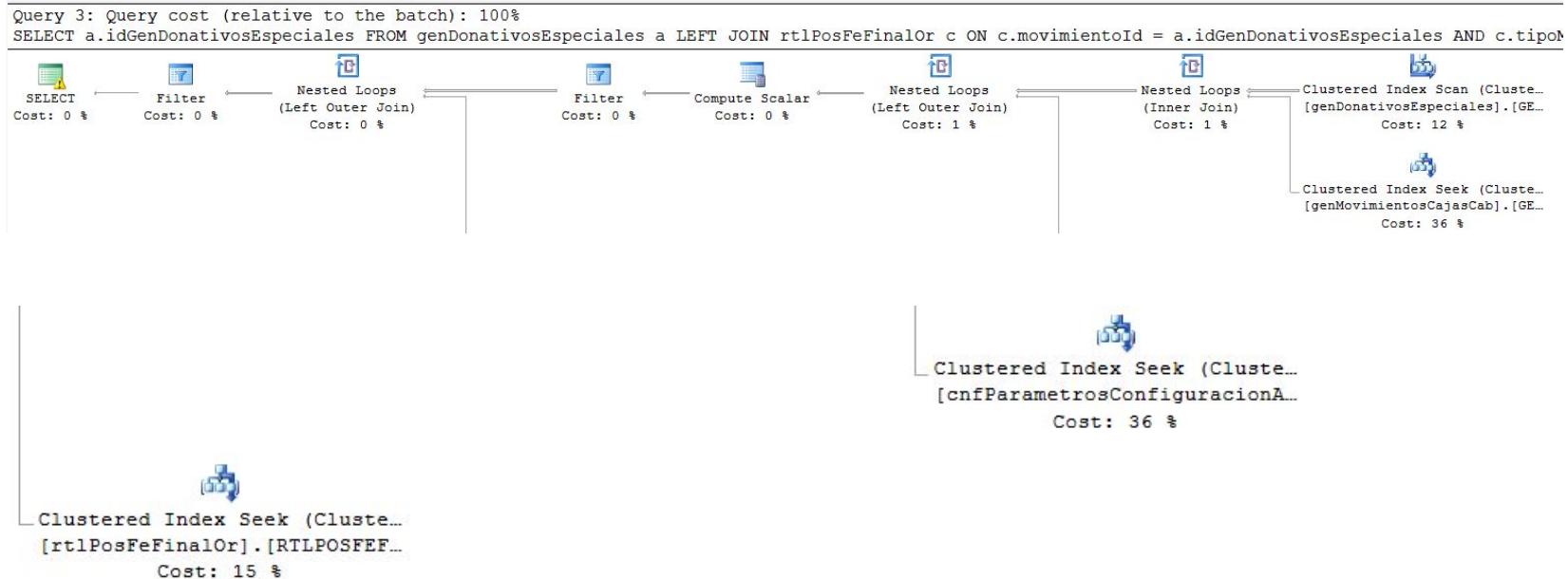
Resultados de las pruebas con límite de paralelismo de 50: Transacciones aplicadas



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 5: Donativos especiales aplicados

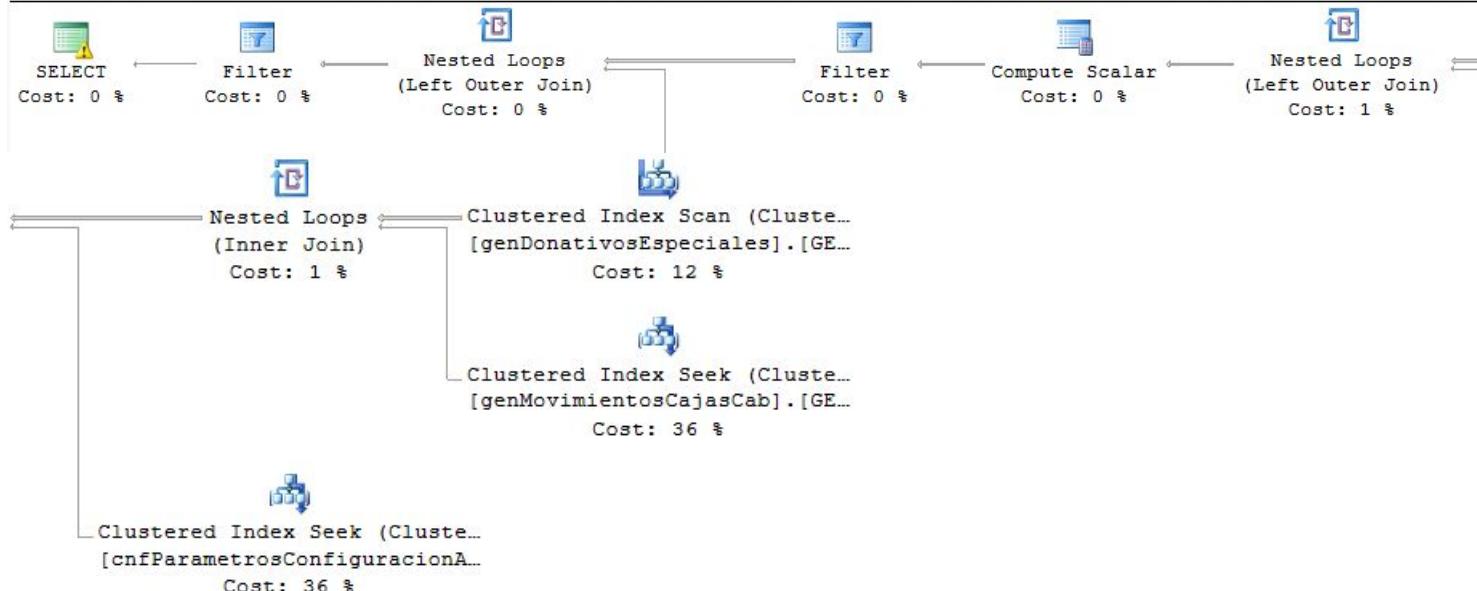


Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 50: Donativos especiales aplicados

Query 3: Query cost (relative to the batch): 100%
SELECT a.idGenDonativosEspeciales FROM genDonativosEspeciales a LEFT JOIN rtlPosFeFinalOr c ON c.mc...

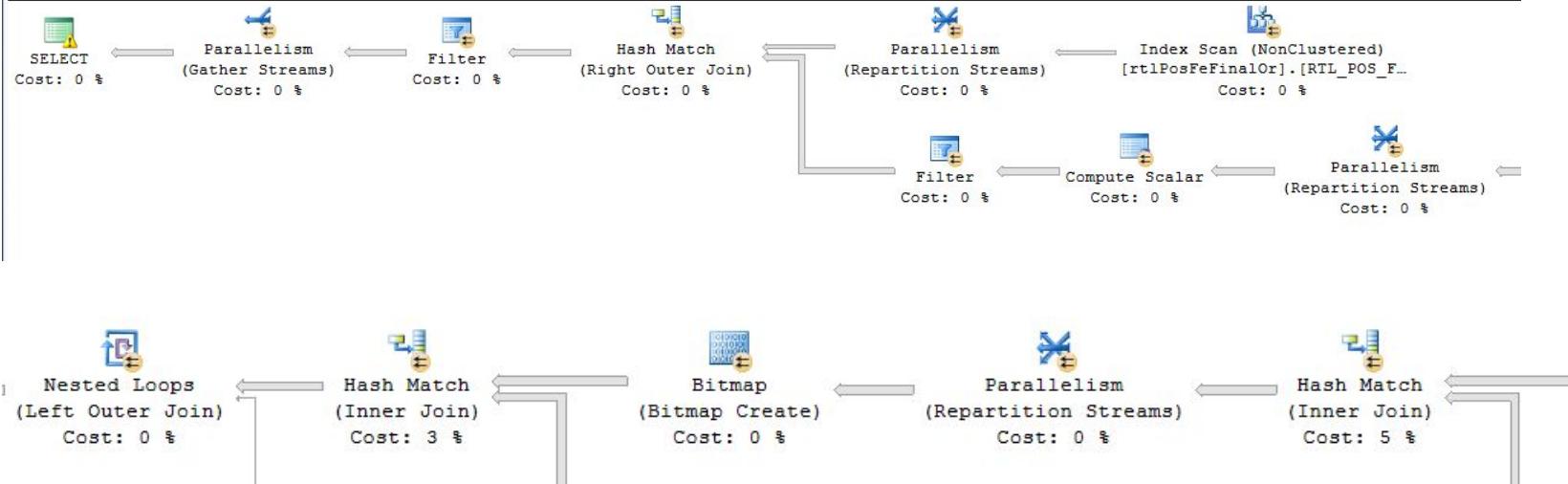


Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 5: Devoluciones aplicadas

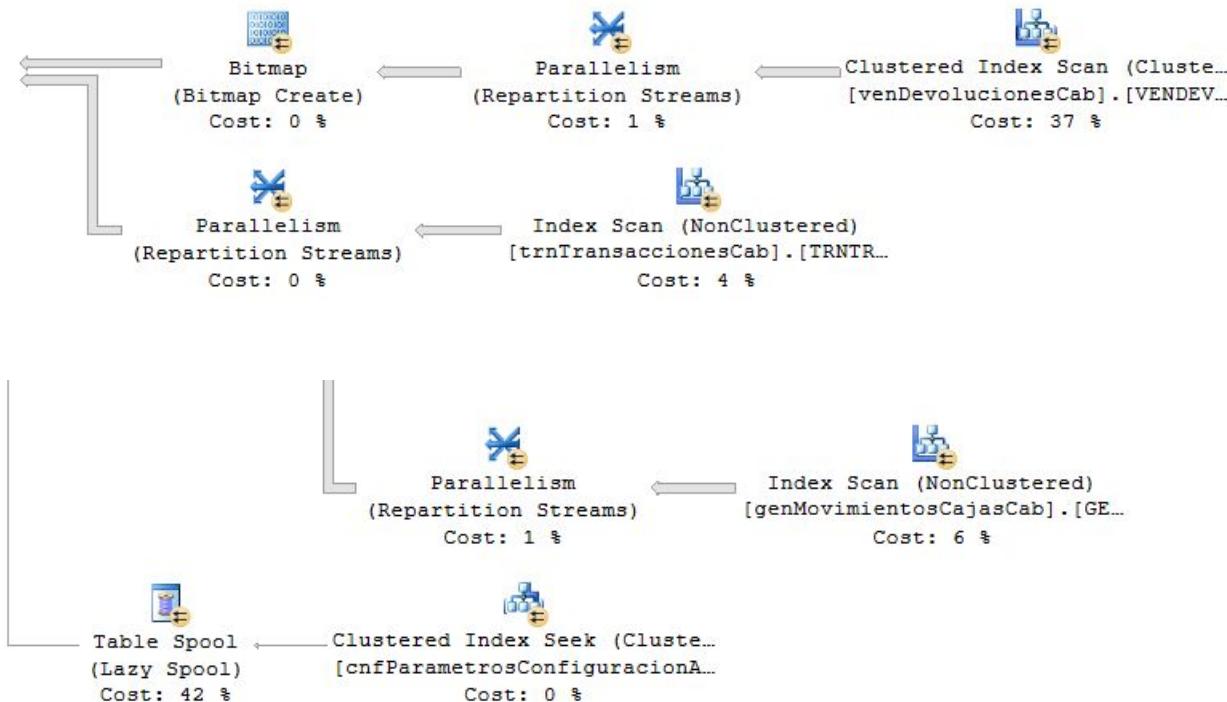
```
Query 3: Query cost (relative to the batch): 100%
SELECT a.idVenDevoluciones FROM venDevolucionesCab a LEFT JOIN rtlPosFeFinalOr c ON c.movimientoId = a.idVenDevoluciones A
Missing Index (Impact 35.3715): CREATE NONCLUSTERED INDEX [<Name of Missing Index, sysname,>] ON [dbo].[venDevolucionesCab]
```



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

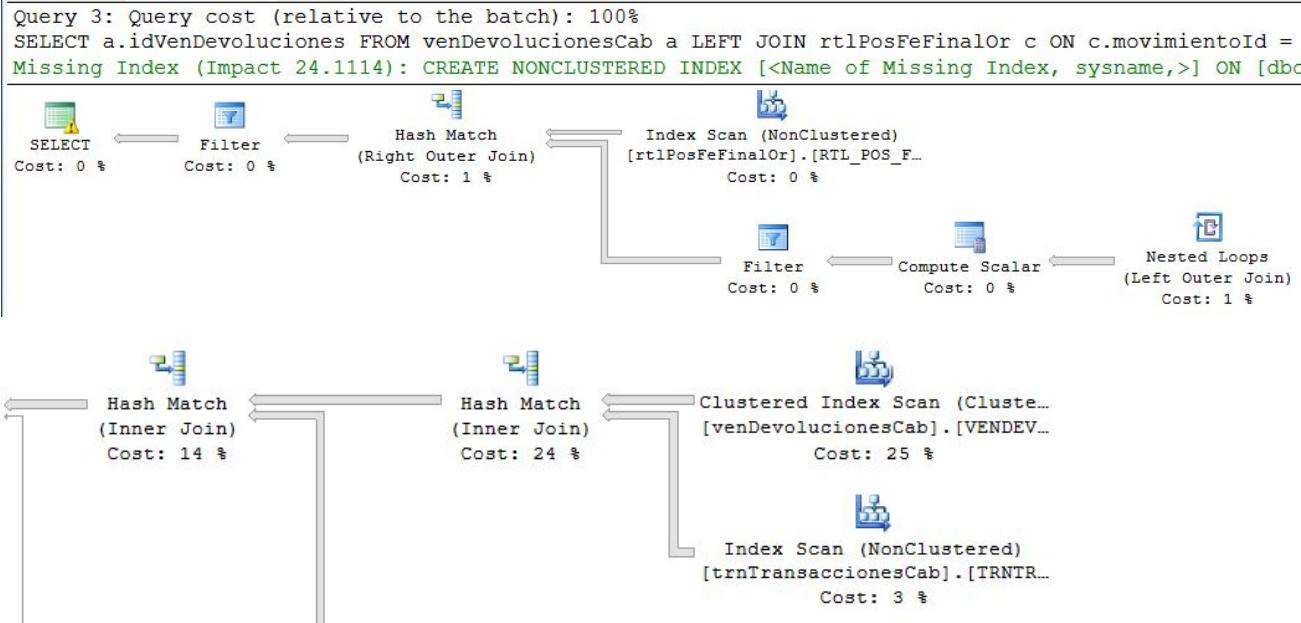
Resultados de las pruebas con límite de paralelismo de 5: Devoluciones aplicadas



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

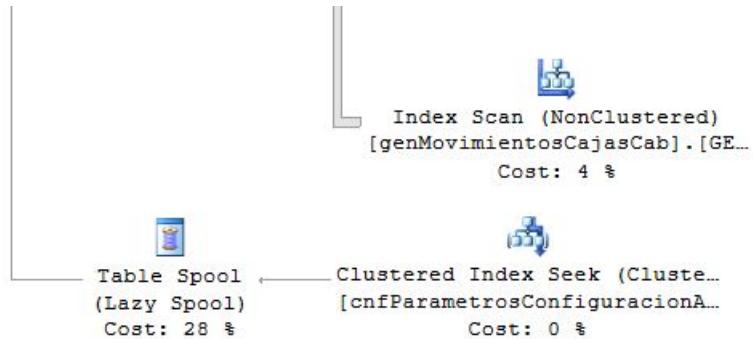
Resultados de las pruebas con límite de paralelismo de 50: Devoluciones aplicadas



Limitar el costo de paralelismo SQL Server

Configuración de parámetro: cost threshold for parallelism

Resultados de las pruebas con límite de paralelismo de 50: Devoluciones aplicadas



Uso de memoria por query

Configuración de parámetro: min memory per query (KB)

Las muestras de resultados se realizaron en base a un grupo de consultas que se realizan en la sucursal.

Se identificaron diversas consultas en procesos base como lo son

1. Promociones por sucursal.
2. Configuraciones asignadas a sucursal.
3. Identificar movimientos como ingresos, devoluciones, donativos especiales para realizar el cierre del día.

Tablas afectadas		
genSucursalesCat	genPromocionesCupon	genMovimientosCajasMsiDet
genAgrupacionesSucursalesCat	cnfConfiguracionAgrupacionSucursalesCnf	cnfParametrosConfiguracionAplicacion
genAsociacionesAgrupacionSucursales	genConfiguracionAgrupacionSucursalesReg	genDonativosEspeciales
genAgrupacionSucursalesPromociones	trnTransaccionesCab	venDevolucionesCab
genPromociones	rtlPosFeFinalOr	sadOrdenSeguimiento
genReglasPrecioPromocion	genMovimientosCajasCab	sadRegistroLiquidacion
genPeriodosGruposPeriodo	trnTransaccionesDet	
genPeriodos	genMovimientosCajasDet	

Uso de memoria por query

Configuración de parámetro: min memory per query (KB)

En cada muestra de pruebas se realizaron las siguientes acciones

1. Validar y confirmar que el valor de la propiedad "min memory per query (KB)" de la base de datos se encuentre en 1024 KB.
2. Ciclo #1, se realizan prueba de consultas detectadas.
3. Toma de valores de uso de cantidades mínimas y máximas de: lecturas físicas y lógicas de páginas, escrituras lógicas que realiza el plan de ejecución.
4. Se realiza configuración de la propiedad "min memory per query" de la base de datos a 2048 KB.
5. Reinicio de servicio de Windows MSSQLSERVER.
6. Ciclo #2, se realizan prueba de consultas detectadas.
7. Toma de valores de uso de cantidades mínimas y máximas de: lecturas físicas y lógicas de páginas, escrituras lógicas que realiza el plan de ejecución.

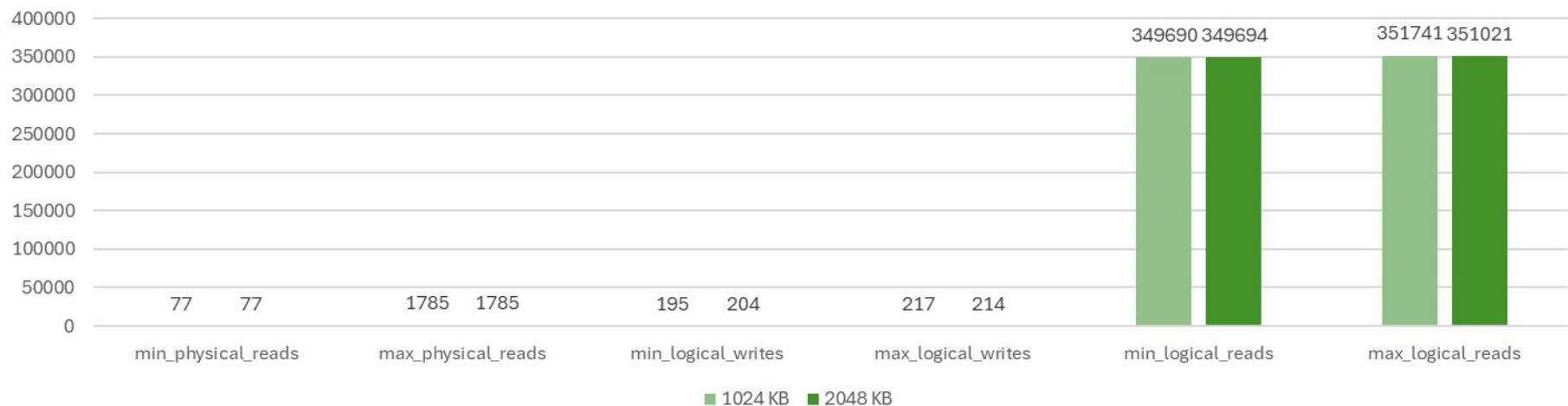
Concepto	Descripción
min_physical_reads	Lectura mínima de páginas de datos desde el disco duro realizada por la misma consulta.
max_physical_reads	Lectura máxima páginas de datos desde el disco duro realizada por la misma consulta.
min_logical_writes	Escritura mínima de páginas de datos desde la caché realizada por la misma consulta.
max_logical_writes	Escritura máxima de páginas de datos desde la caché realizada por la misma consulta.
min_logical_reads	Lectura mínima de páginas de datos desde la caché realizada por la misma consulta.
max_logical_reads	Lectura máxima de páginas de datos desde la caché realizada por la misma consulta.

Uso de memoria por query

Configuración de parámetro: min memory per query (KB)

Resultados de las pruebas

Promociones por sucursal



Uso de memoria por query

Configuración de parámetro: min memory per query (KB)

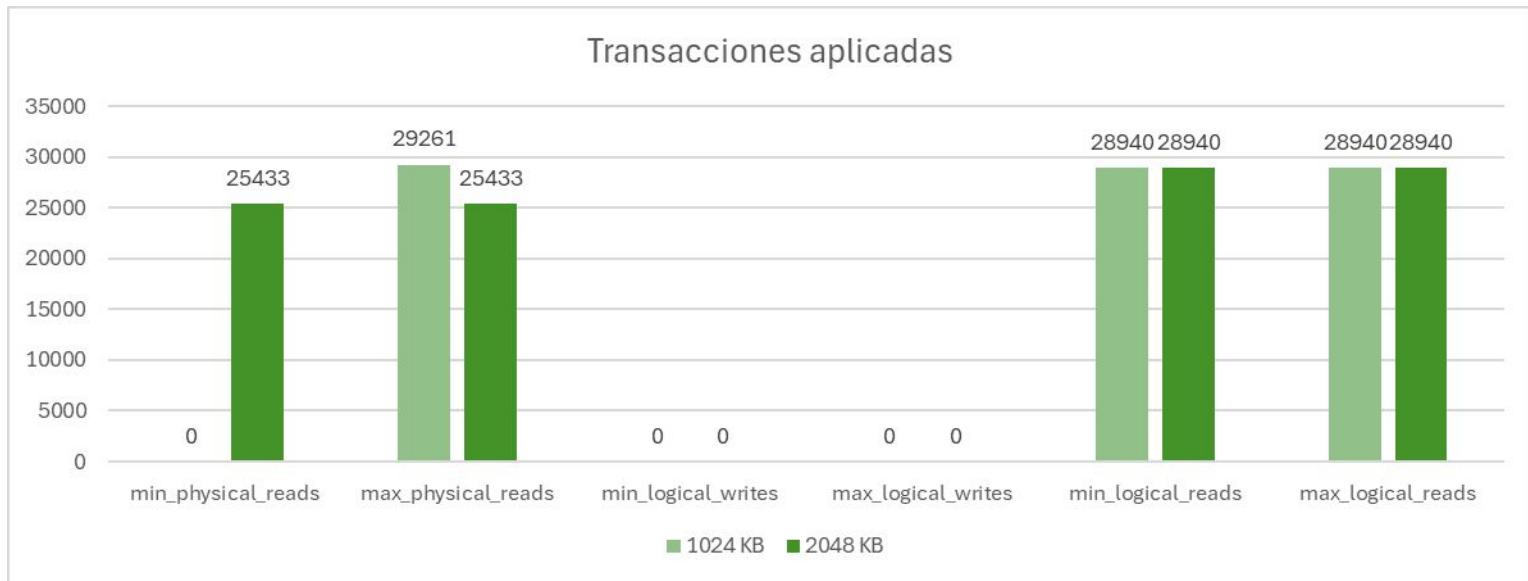
Resultados de las pruebas



Uso de memoria por query

Configuración de parámetro: min memory per query (KB)

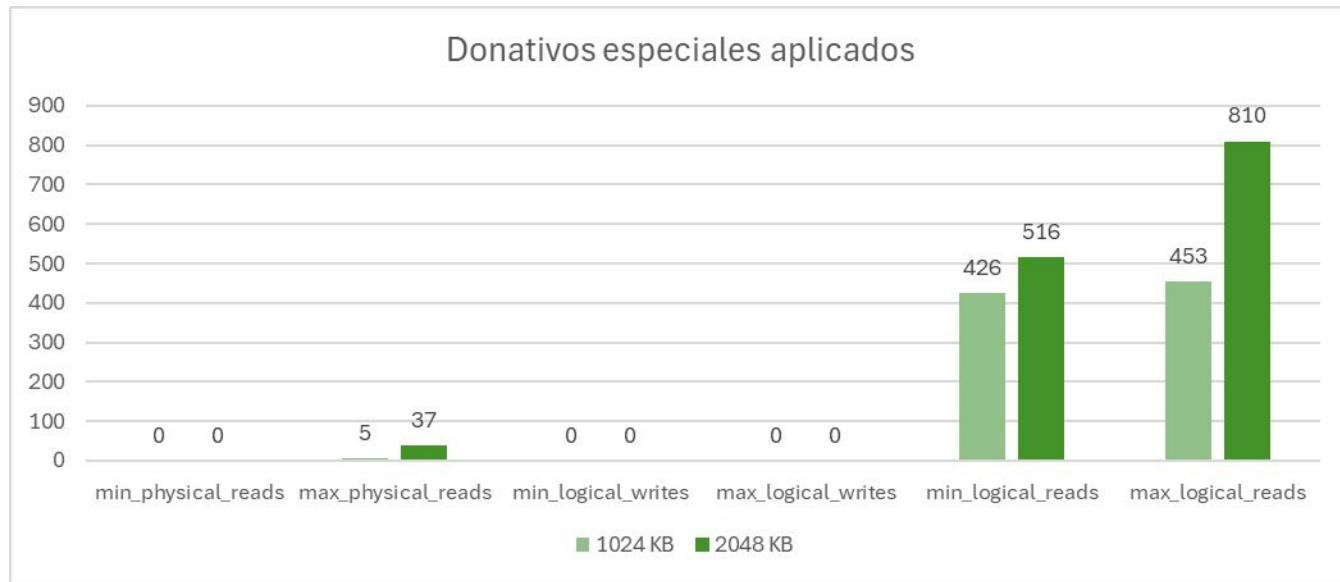
Resultados de las pruebas



Uso de memoria por query

Configuración de parámetro: min memory per query (KB)

Resultados de las pruebas



Uso de memoria por query

Configuración de parámetro: min memory per query (KB)

Resultados de las pruebas



Crecimiento en base de datos

Configuración de parámetro: FILEGROWTH

En cada muestra de pruebas se realizaron las siguientes acciones

1. Validar y confirmar que el valor de la propiedad FILEGROWHT de la base de datos se encuentre en 1MB.
2. Ciclo #1, se realizan 50,000 inserciones en las tablas afectadas.
3. Toma de valores de uso de disco posteriores a la inserción de registros.
4. Ciclo #2, se realizan 50,000 inserciones en las tablas afectadas.
5. Toma de valores de uso de disco posteriores a la inserción de registros.
6. Se realiza configuración de la propiedad FILEGROWHT de la base de datos a 100MB.
7. Reinicio de servicio de Windows MSSQLSERVER.
8. Ciclo #3, se realizan 50,000 inserciones en las tablas afectadas.
9. Toma de valores de uso de disco posteriores a la inserción de registros.
10. Ciclo #4, se realizan 50,000 inserciones en las tablas afectadas.
11. Toma de valores de uso de disco posteriores a la inserción de registros.

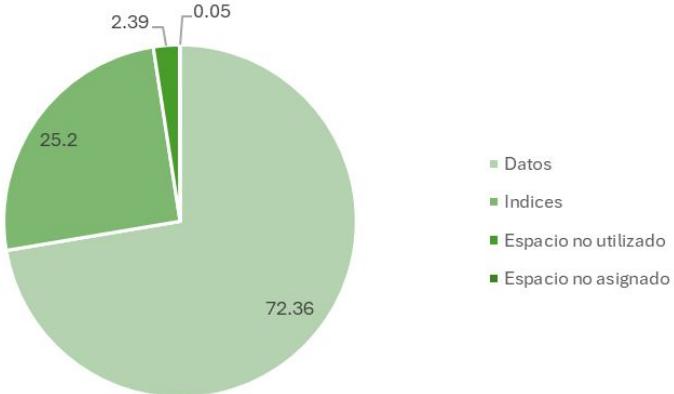
Crecimiento en base de datos

Configuración de parámetro: FILEGROWTH

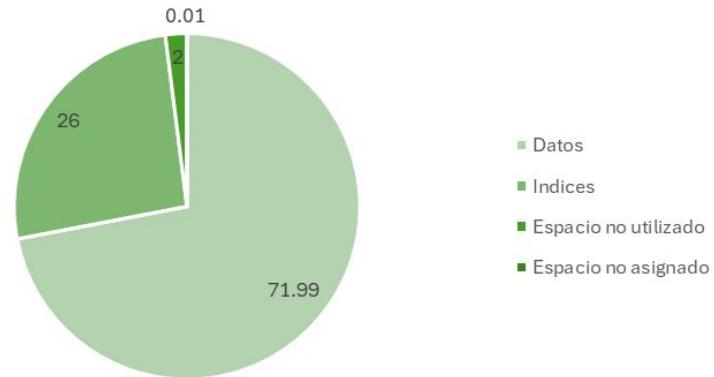
Resultados de cada ciclo de pruebas

Estadística inicial con FILEGROWHT de 1 Mb.

Ciclo #1



Ciclo #2



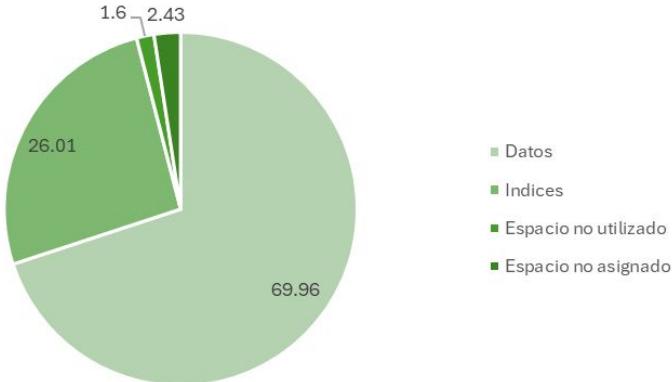
Crecimiento en base de datos

Configuración de parámetro: FILEGROWTH

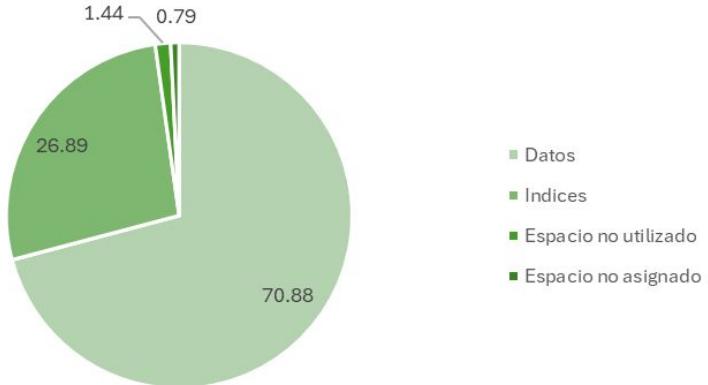
Resultados de cada ciclo de pruebas

Estadística final con FILEGROWHT de 100 Mb.

Ciclo #3



Ciclo #4



Fragmentación en base de datos

Configuración de parámetro: AUTO_SHRINK

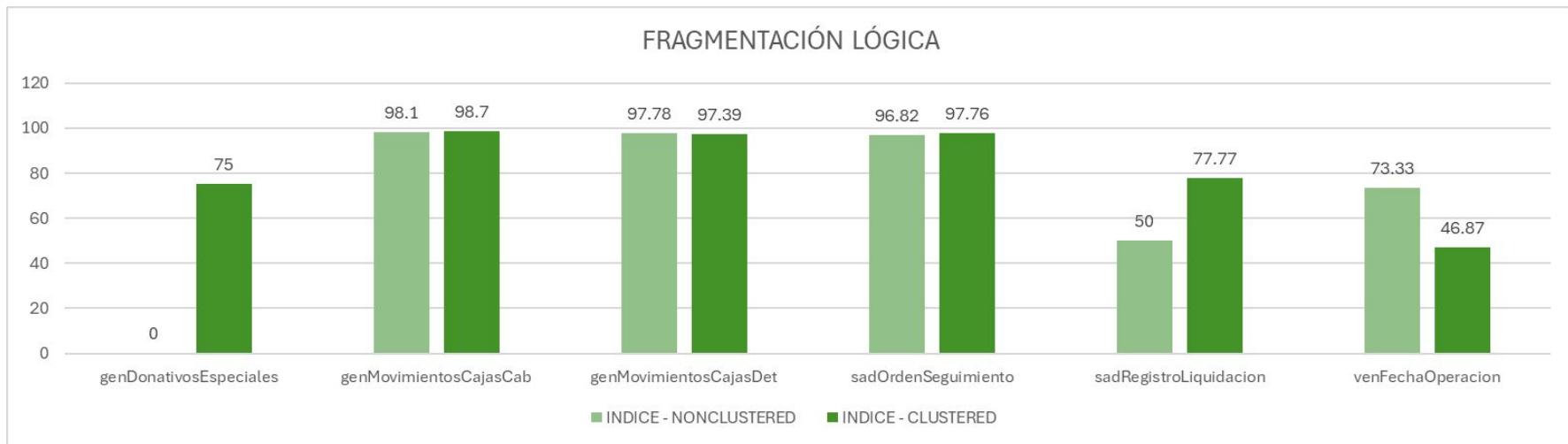
Fueron monitoreadas diversas bases de datos de sucursales y tenían deshabilitada la propiedad AUTO_SHRINK, lo cual es correcto. En caso de ser habilitado, SQL Server intentara fragmentar los índices CLUSTERED Y NONCLUSTERED por lo que esta tarea consumiría mayor rendimiento del equipo de cómputo. No es recomendable habilitarla, para ello se recomienda utilizar un mantenimiento con cierta periodicidad de fragmentación.

Fueron evaluados dos aspectos: fragmentación lógica y densidad de la página.

Fragmentación en base de datos

Configuración de parámetro: AUTO_SHRINK

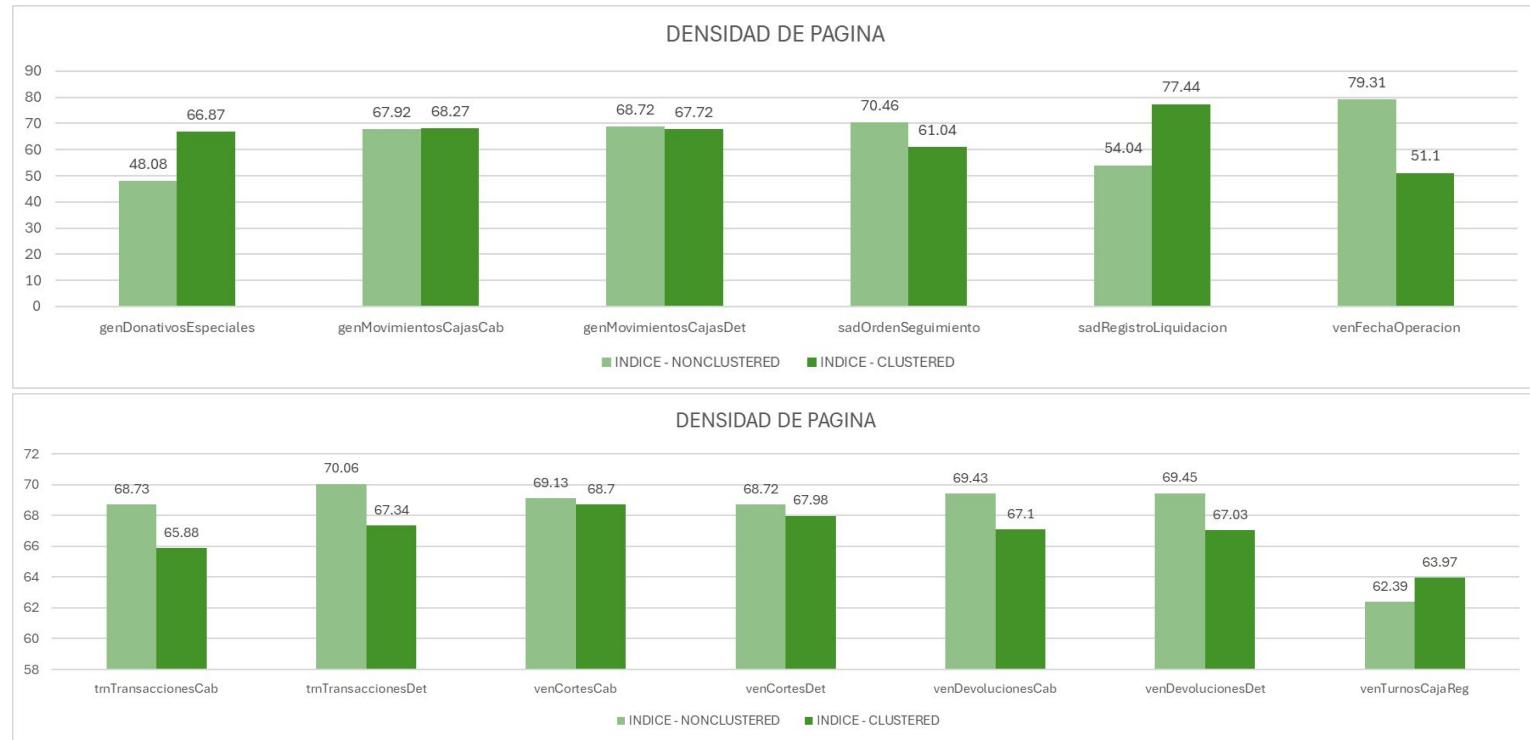
Fragmentación lógica degradada por tabla, páginas de un índice que NO están ordenadas de forma correcta. Entre menor el valor es más degrada la fragmentación.



Fragmentación en base de datos

Configuración de parámetro: AUTO_SHRINK

Degrado de densidad de la página por tabla y tipo de índice. Entre menor su valor el espacio sin utilizar es mayor.



Fragmentación en base de datos

Configuración de parámetro: AUTO_SHRINK

Fueron monitoreadas diversas bases de datos de sucursales y tenían deshabilitada la propiedad AUTO_SHRINK, lo cual es correcto. En caso de ser habilitado, SQL Server intentaría fragmentar los índices CLUSTERED Y NON-CLUSTERED por lo que esta tarea consumiría mayor rendimiento del equipo de cómputo. No es recomendable habilitarla, para ello se recomienda utilizar un mantenimiento con cierta periodicidad de fragmentación.

Se recolectaron estadísticas que muestran aquellos índices que generan una densidad baja de páginas esto quiere decir que tienen espacio vacío (sin usar). Entre mayor densidad de página el espacio hay menos espacio vacío.

Otro aspecto es la fragmentación lógica, entre menos exista y el uso de la tabla es constante en consultas afectaría el rendimiento, es decir existirá lentitud. Ocurre porque al momento de insertar y/o actualizar información hace que los índices se dispersen en diferentes páginas y estos no sigan un orden.

Notas:

No realizo algún cambio en el parámetro ya que las sucursales tienen la fragmentación automática desactivada.

La tarea de fragmentar a realizar es de mantenimiento de forma periódica previa a un full backup la base de datos Farmax_local.

Conclusiones

Las siguientes conclusiones son derivadas a los resultados previos y posterior al ajuste a cada parámetro.

Parámetro: min memory per query (KB)

La diferencia entre las lecturas y escrituras de página previa y posterior al cambio no fueron relevantes. Esto a consecuencia de que las consultas hacia la base de datos utilizan índices tipo NON-CLUSTERED, existe espacio en páginas sin usar entre los índices.

Parámetro: min server memory (MB)

La diferencia entre la memoria física y lógica utilizada previa y posterior al cambio será más notoria cuando exista una volumetría constante de operaciones en la base de datos de sucursal, ya que la liberación de memoria del servidor se realiza constante y controlada, es decir existe una reserva mínima de memoria para las transacciones.

Parámetro: cost threshold for parallelism

No existe diferencia entre el costo que genera el plan de ejecución, esto se puede ser notorio si se disminuye la cantidad de procesadores del equipo. Se encontró alto costo que genera actualmente las consultas y es debido al uso constante de índices tipo NON-CLUSTER, también se generan varios table full scan en consultas, además el uso de tablas temporales en SQL Server.

Parámetro: AUTO_SHRINK

La auto fragmentación se encuentra desactivada lo cual es correcto, lo recomendable es realizar un mantenimiento de fragmentación y ajustar el factor de relleno de los índices para aprovechar el espacio de las páginas.

Parámetro: FILEGROWTH

Existió una diferencia y el espacio sin utilizar disminuyó, aprovechando los índices para almacenar los datos en las páginas.

Tablas e Índices: No se realizaron ajustes ya que los índices NON-CLUSTERED actuales se deben modificar y actualizar su factor de relleno, Estos índices son utilizados en consultas.



CODIFICANDO EL FUTURO



Gracias por
su atención

Extremadura 53, piso 4,
Insurgentes Mixcoac, C.P. 03920
Benito Juárez, CDMX, México
+52 55 6378 3277
www.gettechnologiesmx.com