

Version Inicial Reto 3

(700 rows affected)

(1 row affected)

Tiempos de ejecución de SQL Server:

Tiempo de CPU = 110 ms, tiempo transcurrido = 301 ms.

Completion time: 2023-05-03T20:28:51.1408440-06:00

|

--V1

CREATE NONCLUSTERED INDEX IX_transactionLogs_actorID ON transactionLogs (actorID);

La columna actorID de la tabla transactionLogs es utilizada para unir la tabla con la tabla actores. Para mejorar la velocidad de esta unión, se sugiere agregar un índice no agrupado en la columna actorID.

(700 rows affected)

(1 row affected)

Tiempos de ejecución de SQL Server:

Tiempo de CPU = 109 ms, tiempo transcurrido = 172 ms.

Completion time: 2023-05-03T20:29:44.9156805-06:00

|

--V2

CREATE NONCLUSTERED INDEX IX_transactionLogs_Type_Date ON transactionLogs (transactionTypeID, posttime)

Crear un índice no agrupado en transactionLogs en las columnas transactionTypeID y posttime, en ese orden. El uso de este índice ayudará a acelerar la búsqueda de los registros que cumplen con los criterios del WHERE.

(700 rows affected)

(1 row affected)

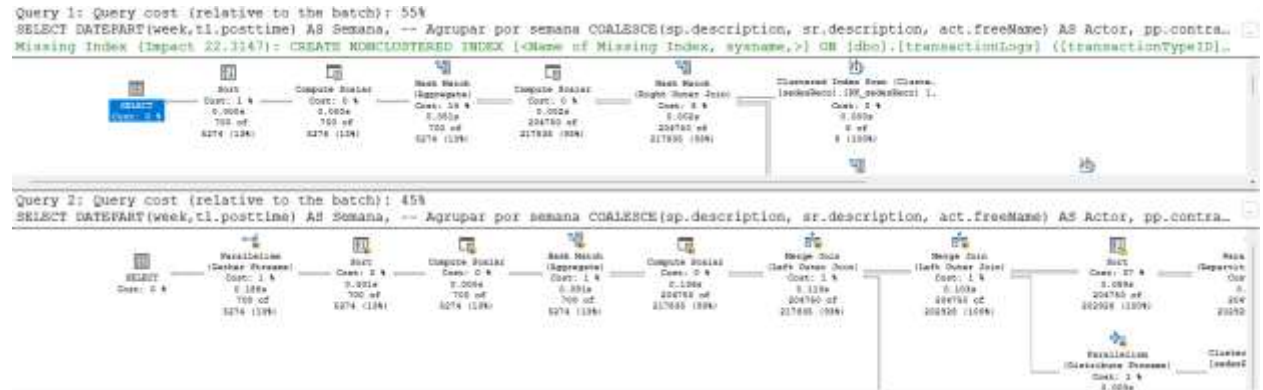
Tiempos de ejecución de SQL Server:

Tiempo de CPU = 109 ms, tiempo transcurrido = 168 ms.

Completion time: 2023-05-03T20:41:52.1335219-06:00

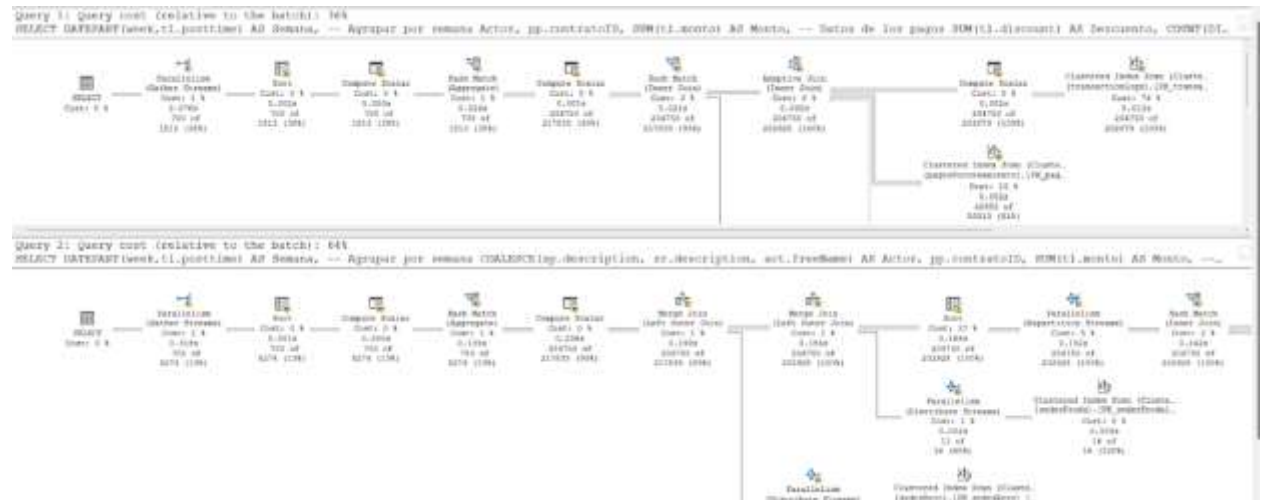
--V3

Se modifica la forma de hacer los left join por medio del merge join. Este merge join combina dos conjuntos de datos ordenados en función de una columna común (clave de unión). En el caso de un left join, se ordena tanto la tabla de la izquierda como la tabla de la derecha por la columna de unión, y luego se realiza una combinación secuencial de ambas tablas en función de la columna de unión. Esto es más rápido y eficiente en términos de uso de memoria que un hash join que requiere la creación de una tabla hash en la memoria para la tabla más pequeña.



--V4

Se hace una optimización en la forma de tomar los actores. Se hace un subquery conectando los actores unirlos con los transactions logs.



Unidad de workload	Explicacion	Norma
<div><div><div>Clustered Index Scan (Clustered)</div><div>Scanning a clustered index, entirely or only a range.</div></div><div><div>Physical Operation</div><div>Clustered Index Scan</div></div><div><div>Logical Operation</div><div>Clustered Index Scan</div></div><div><div>Actual Execution Mode</div><div>Batch</div></div><div><div>Estimated Execution Mode</div><div>Batch</div></div><div><div>Storage</div><div>RowStore</div></div><div><div>Actual Number of Rows Read</div><div>250990</div></div><div><div>Actual Number of Rows for All Executions</div><div>204750</div></div><div><div>Actual Number of Batches</div><div>228</div></div><div><div>Estimated I/O Cost</div><div>2.26461</div></div><div><div>Estimated Operator Cost</div><div>2.53983 (17%)</div></div><div><div>Estimated Subtree Cost</div><div>2.53983</div></div><div><div>Estimated CPU Cost</div><div>0.775223</div></div><div><div>Estimated Number of Executions</div><div>1</div></div><div><div>Number of Executions</div><div>1</div></div><div><div>Estimated Number of Rows for All Executions</div><div>202879</div></div><div><div>Estimated Number of Rows Per Execution</div><div>202879</div></div><div><div>Estimated Number of Rows to be Read</div><div>250300</div></div><div><div>Estimated Row Size</div><div>45 B</div></div><div><div>Actual Rebinds</div><div>0</div></div><div><div>Actual Rewinds</div><div>0</div></div><div><div>Ordered</div><div>False</div></div><div><div>Node ID</div><div>13</div></div></div> <div><div>Predicate</div><div>[esenciaVerde].[dbo].[transactionLogs].[transactionTypeID] as [t]. [transactionTypeID]=1) AND datepart(week,[esenciaVerde].[dbo]. [transactionLogs].[posttime] as [t].[posttime])<10)</div></div> <div><div>Object</div><div>[esenciaVerde].[dbo].[transactionLogs].[PK_transactionLogs] [t]</div></div> <div><div>Output List</div><div>[esenciaVerde].[dbo].[transactionLogs].[referenceID] [esenciaVerde]. [dbo].[transactionLogs].[monto] [esenciaVerde].[dbo]. [transactionLogs].[posttime] [esenciaVerde].[dbo]. [transactionLogs].[actorID] [esenciaVerde].[dbo]. [transactionLogs].[discount]</div></div>	<p>Se recorre todas las filas de transactionLogs, filtrando las que sean del transactionType = 1 y que su fecha sea mayor a la semana 10, todo esto de manera lineal.</p>	<ul style="list-style-type: none">Crear un indice no agrupado en transactionLogs en las columnas transactionTypeID y posttime, en ese orden.
<div><div><div>Hash Match</div><div>Use each row from the top input to build a hash table, and each row from the bottom input to probe into the hash table, outputting all matching rows.</div></div><div><div>Physical Operation</div><div>Hash Match</div></div><div><div>Logical Operation</div><div>Inner Join</div></div><div><div>Actual Execution Mode</div><div>Batch</div></div><div><div>Estimated Execution Mode</div><div>Batch</div></div><div><div>Actual Number of Rows for All Executions</div><div>204750</div></div><div><div>Actual Number of Batches</div><div>228</div></div><div><div>Estimated Operator Cost</div><div>0.771295 (11%)</div></div><div><div>Estimated I/O Cost</div><div>0</div></div><div><div>Estimated CPU Cost</div><div>0.551232</div></div><div><div>Estimated Subtree Cost</div><div>3.32694</div></div><div><div>Number of Executions</div><div>1</div></div><div><div>Estimated Number of Executions</div><div>1</div></div><div><div>Estimated Number of Rows for All Executions</div><div>202879</div></div><div><div>Estimated Number of Rows Per Execution</div><div>202879</div></div><div><div>Estimated Row Size</div><div>49 B</div></div><div><div>Actual Rebinds</div><div>0</div></div><div><div>Actual Rewinds</div><div>0</div></div><div><div>Node ID</div><div>10</div></div></div> <div><div>Output List</div><div>[esenciaVerde].[dbo].[transactionLogs].[referenceID] [esenciaVerde]. [dbo].[transactionLogs].[monto] [esenciaVerde].[dbo]. [transactionLogs].[discount] [esenciaVerde].[dbo]. [actores].[referenceID] [esenciaVerde].[dbo].[actores].[objectTypeID] [esenciaVerde].[dbo].[actores].[treeName] Expr1005</div></div> <div><div>Hash Keys Probe</div><div>[esenciaVerde].[dbo].[transactionLogs].[actorID]</div></div>	<p>Se hace la unión de los actores a cada uno de los transactionLogs mediante una función de hash que utiliza el actorID como llave.</p>	<ul style="list-style-type: none">Para mejorar la velocidad de esta unión, se sugiere agregar un indice no agrupado en la columna actorID.

Hash Match	
Use each row from the top input to build a hash table, and each row from the bottom input to probe into the hash table, outputting all matching rows.	
Physical Operation	Hash Match
Logical Operation	Right Outer Join
Actual Execution Mode	Batch
Estimated Execution Mode	Batch
Actual Number of Rows for All Executions	204750
Actual Number of Batches	228
Estimated Operator Cost	0.5510904 (8%)
Estimated I/O Cost	0
Estimated Subtree Cost	5.33812
Estimated CPU Cost	0.551082
Estimated Number of Executions	1
Number of Executions	1
Estimated Number of Rows for All Executions	202928
Estimated Number of Rows Per Execution	202928
Estimated Row Size	77.8
Actual Rebinds	0
Actual Rewinds	0
Node ID	8
Output List	
[esenciaVerde].[dbo].[transactionLog].[monto]; [esenciaVerde].[dbo].[transactionLog].[discount]; [esenciaVerde].[dbo].[pagosProcesamiento].[pagoProcesamientoID]; [esenciaVerde].[dbo].[pagosProcesamiento].[contratoID]; [esenciaVerde].[dbo].[actores].[referenceID]; [esenciaVerde].[dbo].[actores].[objectTypeID]; [esenciaVerde].[dbo].[actores].[firstName]; [esenciaVerde].[dbo].[sedeProdu].[description]; Expr1005	
Hash Keys Probe	
[esenciaVerde].[dbo].[actores].[referenceID]	
Probe Residual	
[esenciaVerde].[dbo].[sedeProdu].[sedeProduID] as [sp]; [sedeProduID]=[esenciaVerde].[dbo].[actores].[referenceID] as [act].[referenceID] AND [esenciaVerde].[dbo].[actores].[objectTypeID] as [act].[objectTypeID]=1	

Tanto en este como en el otro left join se busca asociar el actor con su respectiva sede, ya sea productor o recolector, esto mediante hash tables que usan keys los referenceID de los actores

- El uso de MERGE JOIN en lugar de HASH JOIN en un LEFT JOIN puede mejorar el rendimiento de la consulta al reducir la necesidad de memoria y acelerar el proceso de unión de las tablas.
- Además de esto se utiliza un subquery para refactorizar el orden de la consulta y evitar hacer tres recorridos con tantos elementos.