

CORRUPCIÓN EN BASE DE DATOS MS SQL SERVER

Tal vez un usuario haya informado que algunos datos no están disponibles, tal vez haya recibido un mensaje de error en un informe que no debería tener errores, tal vez valore su trabajo y tenga programado CHECKDB, que detectó esto temprano. A veces la corrupción de la base de datos te sorprende y hay que solucionarlo. No es un proceso divertido, pero sí esencial.

Causa



Una de las causas más comunes de corrupción es una interrupción en la cadena de páginas de datos en sus índices. Los datos almacenados en su SQL Server se almacenan en páginas de 8kb. Cada una de estas páginas tiene metadatos almacenados, incluidos (entre otros) el ID de la página y también los ID de la página anterior y siguiente. Esto permite a SQL Server garantizar la coherencia de los resultados. Cuando se leen datos de una tabla o índice que tiene una interrupción en esta cadena, la consulta fallará y obtendrá un error que muestra que existe una posible corrupción.

Cuando esto sucede, tenemos que encontrar el alcance de esta corrupción, ya que algunos casos son mucho más fáciles de abordar que otros.

El mensaje de error recibido mostrará qué tabla informa daños y, a veces, también le dará el nombre del índice. Si no es así, entonces debemos encontrar qué índice de esta tabla está potencialmente afectado.

La forma más sencilla de hacerlo es obtener una lista de sus índices. Puede hacer esto en la GUI de SSMS o utilizar la siguiente consulta (reemplace **NombreTabla** con el nombre real de su tabla)

```
SELECT
  o.name AS TableName
  ,i.name AS IndexName
  ,i.type_desc AS IndexType
FROM sys.indexes i
INNER JOIN sys.objects o ON i.OBJECT_ID = o.OBJECT_ID
WHERE o.name = 'NombreTabla'
```



CORRUPCIÓN EN BASE DE DATOS MS SQL SERVER

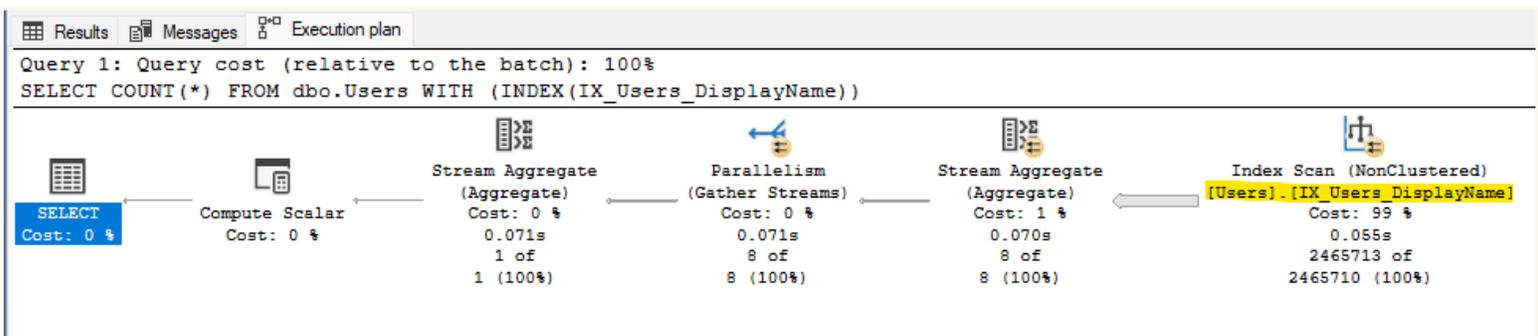
Estoy usando una copia de la base de datos de desbordamiento de pila y mis índices se ven así;

	TableName	IndexName	IndexType
1	Users	PK_Users_Id	CLUSTERED
2	Users	IX_Users_DisplayName	NONCLUSTERED
3	Users	IX_Users_Reputation_Filtered	NONCLUSTERED

Luego, haremos una consulta de “select *” simple para cada índice individualmente. Para hacer esto, usaremos una sugerencia de índice (si te veo haciendo esto en el código de producción, tendremos palabras):

```
SELECT COUNT(*)  
FROM dbo.Users  
WITH (INDEX(IX_Users_DisplayName))
```

Si observa el plan de ejecución de la consulta, puede estar seguro de que está utilizando el índice correcto:



Querrá ejecutar la misma consulta con todos sus índices. Cualquiera con corrupción arrojará un error. Ahí es donde debemos centrarnos.