

شناسایی ایندکس های بدون استفاده در Sql server

برای اینکه اطمینان حاصل کنید که دسترسی به داده تا جایی که ممکن است سریع می باشد، **SQL Server** مانند سیستم های مرتبط دیتابیس دیگر از نمایه سازی (**indexing**) برای یافتن سریع داده استفاده می کند. **SQL Server** ایندکس های مختلفی دارد که ایجاد می شوند، مانند **clustered indexes**، **non-clustered indexes** و **Full Text indexes**.

مزیت داشتن ایندکس های بیشتر این است که اگر ایندکس مناسبی وجود داشته باشد، **SQL Server** می تواند به سرعت به داده دسترسی داشته باشد. عیب داشتن ایندکس های زیاد نیز این است که **SQL Server** باید همه ی این ایندکس ها را حفظ کند که می تواند سرعت کار را پایین آورد و ایندکس ها نیز نیاز به فضای ذخیره سازی بیشتری دارند. بنابراین همانطور که می بینید **indexing** در اجرا هم می تواند مفید و هم مضر باشد.

در این بخشبر روی چگونگی شناسایی ایندکس های موجود که استفاده نمی شوند، تمرکز می کنیم که چنین ایندکس هایی برای بهبود اجرا قابل حذف می باشند و فضای ذخیره سازی مورد نیاز نیز به این شکل کاهش می یابد.

توضیح

وقتی **SQL Server 2005** معرفی شد، **Dynamic Management Views (DMVs)** را اضافه کرد که به شما اجازه می دهد تا یک دیدگاه وسیع تر نسبت به آنچه در یک **SQL Server** در حال انجام است، داشته باشید. یکی از این موارد توانایی مشاهده ی چگونگی استفاده از ایندکس هاست. دو **DMV** وجود دارند که در مورد آنها بحث خواهیم کرد. دقت داشته باشید که این ویوها داده ی جمع شده را ذخیره می کنند، بنابراین وقتی **SQL Server** دوباره اجرا می شود، شمارنده ها به صفر برمی گردند. بنابراین از این مسئله در هنگام نظارت بر استفاده از ایندکس خود آگاه باشید.

DMV - sys.dm_db_index_operational_stats

این **DMV** به شما اجازه می دهد تا اطلاعات مربوط به وارد کردن، آپدیت کردن و یا حذف را برای یک ایندکس از جهات مختلف مشاهده کنید. اساساً میزان تلاش برای برقراری ایندکس بر اساس تغییرات داده را ارائه می دهد.

اگر شما جدول را تنظیم کرده و همه ی ستون ها را گزارش دهید، خزوجی ممکن است گیج کننده باشد. بنابراین **query** زیر روی تعداد کمی از ستون های کلیدی تمرکز می کند. برای یادگیری بیشتر در مورد همه ی ستون ها می توانید **Books Online** را چک کنید:

```
SELECT OBJECT_NAME(A.[OBJECT_ID]) AS [OBJECT NAME],  
I.[NAME] AS [INDEX NAME],
```

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

<http://www.tahlildadeh.com/>

```

A.LEAF_INSERT_COUNT,
A.LEAF_UPDATE_COUNT,
A.LEAF_DELETE_COUNT
FROM SYS.DM_DB_INDEX_OPERATIONAL_STATS (db_id(),NULL,NULL,NULL) A
INNER JOIN SYS.INDEXES AS I
ON I.[OBJECT_ID] = A.[OBJECT_ID]
AND I.INDEX_ID = A.INDEX_ID
WHERE OBJECTPROPERTY(A.[OBJECT_ID], 'IsUserTable') = 1

```

در زیر تعداد **Insert**، **Update** و **Delete** را که برای هر ایندکس اتفاق می افتد، مشاهده می کنیم. این جدول نشان می دهد که **SQL Server** برای برقراری ایندکس چقدر کار باید انجام دهد.

	OBJECT NAME	INDEX NAME	LEAF_INSERT_COUNT	LEAF_UPDATE_COUNT	LEAF_DELETE_COUNT
1	ContactCreditCard	PK_ContactCreditCard_ContactID_CreditCardID	1	0	0
2	CountryRegionCurrency	PK_CountryRegionCurrency_CountryRegionCode_...	13	6036	0
3	Employee	PK_Employee_EmployeeID	0	0	0

DMV - sys.dm_db_index_usage_stats

این **DMV** به شما نشان می دهد که ایندکس برای **query** های یوزر چندبار استفاده شده است. که اگر همه ی ستون ها را تنظیم کنید، ستون های بسیار دیگری وجود دارند که بازگردانده می شوند. برای اطلاعات بیشتر می توانید به **Books Online** مراجعه کنید.

```

SELECT OBJECT_NAME(S.[OBJECT_ID]) AS [OBJECT NAME],
I.[NAME] AS [INDEX NAME],
USER_SEEKS,
USER_SCANS,
USER_LOOKUPS,
USER_UPDATES
FROM SYS.DM_DB_INDEX_USAGE_STATS AS S

```

```

INNER JOIN SYS.INDEXES AS I ON I.[OBJECT_ID] = S.[OBJECT_ID] AND I.INDEX_ID = S.INDEX_ID

```

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

WHERE OBJECTPROPERTY(S.[OBJECT_ID], 'IsUserTable') = 1

AND S.database_id = DB_ID()

در اینجا می توانیم **seeks**، **scans**، **lookups** و **updates** را مشاهده کنیم.

- **Seek** اشاره به تعداد دفعاتی می کند که یک **index seek** برای آن ایندکس اتفاق می افتد. این روش سریعترین راه برای دسترسی به داده می باشد، بنابراین روش خوبی است.
- **Scan** اشاره به تعداد دفعاتی دارد که یک **index scan** برای یک ایندکس اتفاق می افتد، این روش زمانی اتفاق می افتد که چند ردیف داده برای پیدا کردن یک داده جستجو می شوند. شما سعی دارید که از اسکن ها جلوگیری کنید.
- **lookups** اشاره دارد به تعداد دفعاتی که **query** داده را از **clustered index or the heap** خارج می کند. (**clustered index** ندارد). شما سعی در جلوگیری از **Lookups** نیز دارید.
- **Updates** اشاره دارد به تعداد دفعاتی که به خاطر تغییرات داده، ایندکس آپدیت می شود، که باید با اولین **query** بالا هماهنگ باشد.

	OBJECT NAME	INDEX NAME	USER_SEEKS	USER_SCANS	USER_LOOKUPS	USER_UPDATES
9	CountryRegionCurrency	PK_CountryRegionCurrency_CountryRegionCode_Curre...	6036	0	0	6037
10	CustomerAddress	AK_CustomerAddress_rowguid	0	0	0	93
11	CustomerAddress	PK_CustomerAddress_CustomerID_AddressID	93	0	0	93

تشخیص ایندکس های استفاده نشده:

بنابراین براساس خروجی بالا باید روی دومین خروجی **query** بالا تمرکز کنید. اگر ایندکس هایی را مشاهده کردید که در آنها **seek**، **scan** یا **lookup** اتفاق نمی افتد اما آپدیت هایی اتفاق افتاده است، به این معناست که **SQL Server** برای کامل کردن **query** از ایندکس استفاده نکرده، اما نیاز به برقراری ایندکس دارد. توجه داشته باشید که زمانیکه **SQL Server** مجددا شروع به کار کرد، داده ی مربوط به این **DMV** ها مجددا تنظیم می شود، بنابراین اطمینان حاصل کنید که برای یک مدت طولانی داده جمع آوری کرده اید و مشخص کنید که کدام ایندکس ها انتخاب های خوبی برای حذف می باشند.

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

<http://www.tahlildadeh.com/>