

# Эффективность использования индексов Microsoft SQL Server

Одно из ключевых условий эффективности использования Microsoft SQL Server состоит в создании такого набора индексов для таблиц, чтобы любые запросы к таблицам могли бы выполняться эффективно. С увеличением объема данных эффективность использования индексов может снижаться, приводя к увеличению времени выполнения операций по чтению и модификации данных. Microsoft SQL Server имеет свойство автоматического обновления статистики индексов, но для поддержания индексов в актуальном состоянии этого бывает недостаточно, поскольку Microsoft SQL Server не перестраивает индексы автоматически.

## Проблема:

После выполнении интенсивных операций по модификации данных в таблицах базы данных увеличивается время выполнения запросов и операций по модификации данных.

Это обусловлено тем, что при таких операциях происходит модификация индексов, что приводит к их фрагментации и увеличению количества операций ввода-вывода при использовании индексов в процессе выполнения операций чтения и записи данных.

## Решение:

- Регулярная переиндексация таблиц базы данных с помощью команды **DBCC DBREINDEX ( *table\_name* )**.
- Регулярная дефрагментация индексов базы данных с помощью команды **DBCC INDEXDEFRAG(*database\_name*, *table\_name*, *index\_name*)**.

Выбор способа решения этой проблемы зависит от интенсивности операций по модификации таблиц базы данных.

Регулярная переиндексация таблиц базы данных является более эффективной процедурой, однако время выполнения у нее существенно больше. Кроме того, ее выполнение может замедлить работу пользователей, поскольку на время перестроения индекса блокируется доступ к таблице базы данных, индекс которой в данный момент перестраивается.

В отличие от переиндексации, дефрагментация индексов является обычной операцией и не приводит к блокировкам таблиц, поэтому она может выполняться без прерывания работы пользователей. Эта операция использует стандартный механизм транзакций для перемещения страниц индекса. Кроме того, это работает быстрее, чем построение нового индекса. С помощью её можно дефрагментировать и кластерные и не кластерные индексы, что улучшает эффективность доступа к данным, поскольку физический порядок будет соответствовать логическому порядку и уменьшится количество операций ввода-вывода при просмотре индекса.

При высокой интенсивности операций модификации данных, возможно использование комбинации этих методов:

- Выполнение в течении рабочего дня дефрагментации индексов.
- Выполнение переиндексации таблиц базы данных после окончания рабочего дня.

Сочетание этих двух методов позволит поддерживать индексы в актуальном состоянии, независимо от интенсивности операций с базой данных.

Эту процедуру можно автоматизировать, написав скрипт на Transact-SQL, который будет исполняться с требуемой периодичностью с помощью Microsoft SQL Server Agent.

Пример подобного скрипта оформленный в виде хранимой процедуры:

[Копировать в буфер обмена](#)

```
CREATE PROCEDURE DBReindex AS
SET NOCOUNT ON
DECLARE @TableName char(32)
DECLARE SysCur CURSOR FOR SELECT name FROM sysobjects WHERE type='U'
OPEN SysCur
FETCH NEXT FROM SysCur INTO @TableName
```

```
WHILE @@FETCH_STATUS=0 BEGIN
    DBCC DBREINDEX(@TableName)
    FETCH NEXT FROM SysCur INTO @TableName
END
CLOSE SysCur
DEALLOCATE SysCur
```

Аналогичный результат можно получить с помощью **Database Maintenance Plan Wizard** из **SQL Server Enterprise Manager**.

Более подробное описание и рекомендации по использованию этих команд и **Database Maintenance Plan Wizard** можно найти в документации по Microsoft SQL Server.