



Options for Proactively Managing Statistics in SQL Server



sqlbits

March 10, 2022





Erin Stellato

Program Manager, Microsoft



erinstellato@microsoft.com



[@erinstellato](https://twitter.com/erinstellato)



www.sqlskills.com/blogs/erin

Speaker
Trainer
Community Volunteer

I used to be a...
Consultant
Data Platform MVP

Abstract



Have you ever wondered how you could manage statistics beyond having `AUTO_UPDATE_STATISTICS` for your databases enabled, using a maintenance plan, and/or using the default options in Ola Hallengren's script? If so, this session is for you. I'll talk through the method I use to figure out what statistics need frequent updates, which ones have skew, and which ones aren't as critical.

**It's important to understand that in this session,
I will not give you THE answer.
I will share steps to find a good solution,
because there is no ONE solution for all systems.**

How do we typically manage statistics?



Let SQL Server
do it



Just auto update
stats?

Maintenance
Plan



Sledgehammer
approach, but if
you're not a DBA,
and you're not 24x7,
it's fine

Scheduled Job



Best method, as long
as you're **not** using
sp_updatestats

Ola's Options



@UpdateStatistics

NULL is default

ALL, INDEX, COLUMNS

Default for IndexOptimize

```
EXECUTE [dbo].[IndexOptimize]  
@Databases = 'USER_DATABASES',  
@LogToTable = 'Y'
```

It's basically this

```
EXECUTE [dbo].[IndexOptimize]  
@Databases = 'USER_DATABASES',  
@FragmentationLevel1 = 5,  
@FragmentationLevel2 = 30,  
@UpdateStatistics = 'NULL'
```



Ola's Options



@UpdateStatistics

NULL is default
ALL, INDEX, COLUMNS



@OnlyModifiedStatistics

Updates if only 1 row has changed
Doesn't update indexed views



@StatisticsModificationLevel

Specify a percentage, or will use dynamic threshold



@StatisticsSample

Set the sample rate as a percentage (100 is FULLSCAN)

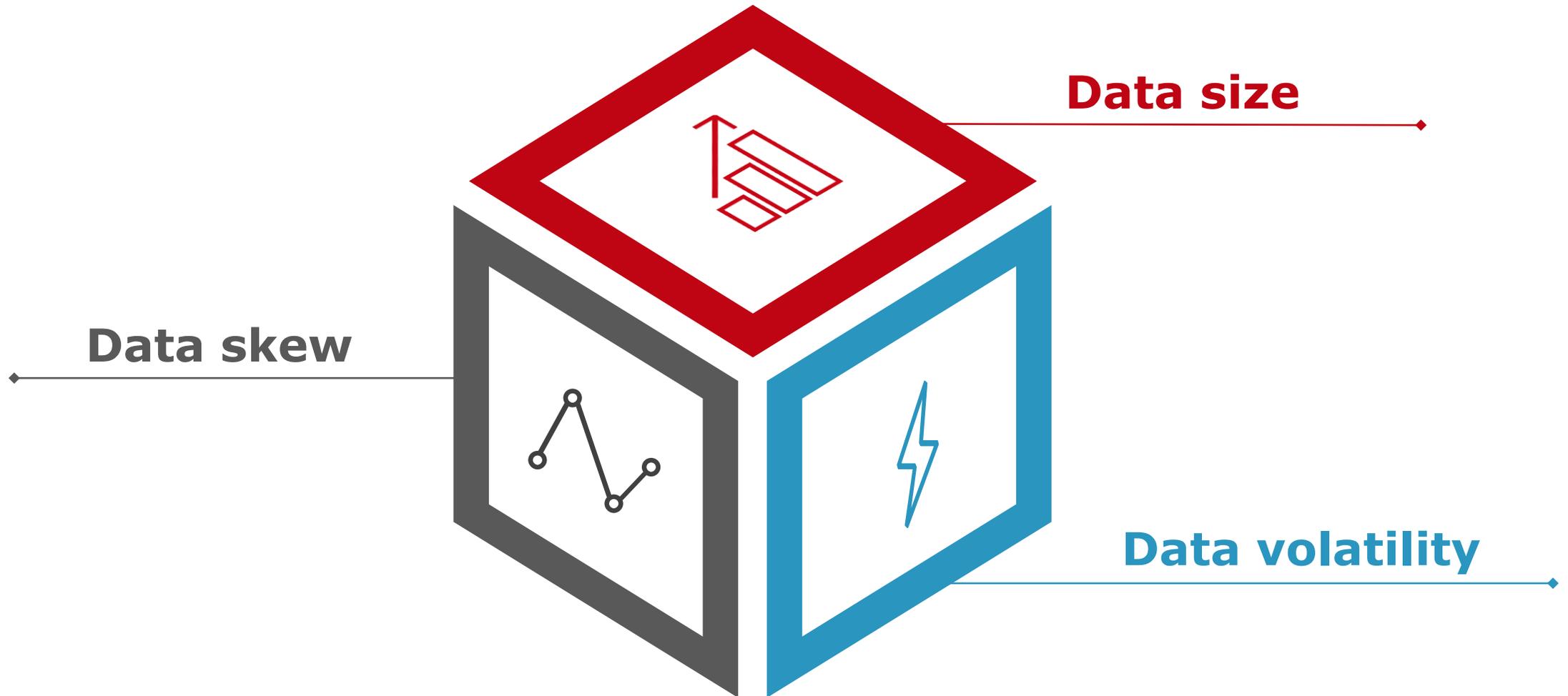
Older Way

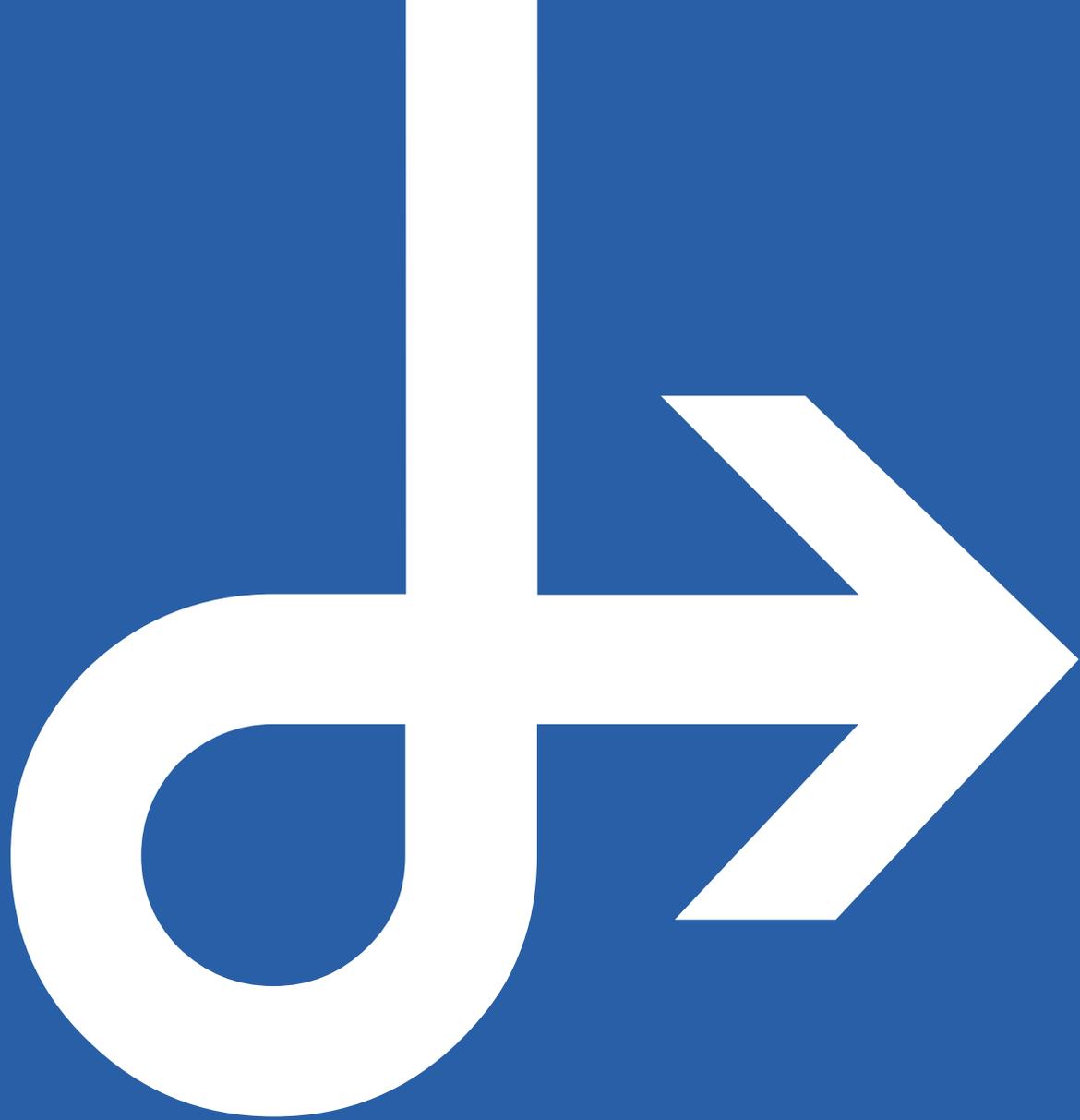
```
EXECUTE dbo.IndexOptimize
@Databases = 'USER_DATABASES',
@FragmentationLow = NULL,
@FragmentationMedium =
'INDEX_REORGANIZE,
INDEX_REBUILD_ONLINE,
INDEX_REBUILD_OFFLINE',
@FragmentationHigh =
'INDEX_REBUILD_ONLINE,
INDEX_REBUILD_OFFLINE',
@FragmentationLevel1 = 5,
@FragmentationLevel2 = 30,
@UpdateStatistics = 'ALL',
@OnlyModifiedStatistics = 'Y'
```

Newer Way

```
EXECUTE dbo.IndexOptimize
@Databases = 'USER_DATABASES',
@FragmentationLow = NULL,
@FragmentationMedium =
'INDEX_REORGANIZE,
INDEX_REBUILD_ONLINE,
INDEX_REBUILD_OFFLINE',
@FragmentationHigh =
'INDEX_REBUILD_ONLINE,
INDEX_REBUILD_OFFLINE',
@FragmentationLevel1 = 5,
@FragmentationLevel2 = 30,
@UpdateStatistics = 'ALL',
@StatisticsModificationLevel= '5'
```

Why one script doesn't work





“DEMO”

Customizing a Solution

Multiple Jobs for Updating Statistics



```
EXECUTE dbo.IndexOptimize
@Databases = 'USER DATABASES',
@FragmentationLow = NULL,
@FragmentationMedium = NULL,
@FragmentationHigh = NULL,
@UpdateStatistics = 'ALL',
@OnlyModifiedStatistics = 'Y',
@LogToTable = 'Y'
```

```
EXECUTE dbo.IndexOptimize
@Databases = 'USER DATABASES',
@FragmentationLow = NULL,
@FragmentationMedium = NULL,
@FragmentationHigh = NULL,
@UpdateStatistics = 'ALL',
@StatisticsModificationLevel = '5',
@LogToTable = 'Y'
```


Thank you!

sqlbits



Feedback

<https://sqlb.it/?6967>

erinstallato@microsoft.com

@erinstallato

Contact



Say hi!