

Clustered Indexes Mia Culpa

Posted At : May 28, 2008 2:59 PM | Posted By : Mark Kruger

Related Categories: MS SQL Server

I have for years espoused the benefits of clustered indexes on MS SQL. Unlike a regular index a "clustered" index represents the actual sort order of the table. It is, therefore, the fastest available type of index. It is my view that some thought should be given to which columns are added to the clustered index. Please note, any indexing plan should include performance metrics coupled with experience. Please don't think I am recommending wholesale changes to any given schema. Having said that, a misunderstanding or misuse of indexing is the *one of the most common cause of performance related problems*. Now back to our discussion of clustered indexing.

Here's the scoop. When you create a primary key using Enterprise Manager or Server Management Studio, MS SQL creates a clustered index by default to accompany the key.

This means, if you wish for some *other* column or columns to be the clustered index you have to alter this arrangement. Now for the last 7 or 8 years I have taught that the way you do this is to drop the primary key, add your clustered index and then reset the primary key. I came to this conclusion in the misty unknown past when I was cutting my teeth with SQL Server 7. Muse reader John Cranston has rightly pointed out that this procedure, while effective, is unnecessary. All that is needed is to **uncheck** the clustered box for the PK index.

You can then safely create your clustered index without interfering with the primary key. Thanks John... and my apologies to anyone who has been unnecessarily dropping their keys with me lo these many years.