BRENT OZAR
UNLIMITED®

Fundamentals of Index Tuning

Recap of what we learned and what to do next.

Selectivity isn't just uniqueness.

It's about what % of an object is matched by:

- Your WHERE clause
- Your JOIN relationships
- The ORDER BY, especially with a TOP

Isolate portions of your query and run 'em separately to see how many rows will match.

Visualize indexes with a query.

Stumped about why SQL Server refuses to use an index or is doing a sort?

Build a query to match the contents of the index, sorted in the same order.

Looking at that output will help you understand why a query will (or won't) use the index, and why a sort will be required after the data comes out of the index.

The first round is the easy button.

The first round of tuning tweaks is very effective:
you can make a huge difference in a couple of hours.

Subsequent rounds are harder,
and produce diminishing returns.

Work hard enough at tuning an application, and you start running out of free/easy options.

Management needs to hear that message:
“We’ve already pushed all the easy buttons.”

The right nonclustered indexes...

1. Reduce PAGEIOLATCH waits because we can grab a few pages from a tiny in-memory index rather than scanning an entire table from disk.
2. Reduce blocking by:
    1. Letting us close transactions faster
    2. Helping us find rows we want to update

The wrong nonclustered indexes

Slow down deletes, updates, and inserts because we have to lock and touch all these extra pages.

Reduce our memory effectiveness because we have to cache all these pages we don't really need (since we're touching them for DUIs.)

Slow down maintenance jobs: backups, checkdb, index rebuilds, stats updates.

My D.E.A.T.H. Method

Dedupe near-identical indexes

Eliminate unused indexes

Add highly needed missing indexes

Tune resource-intensive queries

Heaps often need clustered indexes

Mastering Index Tuning

Just once	**Dedupe** – reduce overlapping indexes with sp_BlitzIndex **Eliminate** – unused indexes with sp_BlitzIndex
Weekly for 1 month	**Add** – badly needed missing indexes with sp_BlitzIndex
Do this only AFTER the easy stuff above	**Tune** – indexes for specific queries with sp_BlitzCache **Heaps** – usually need clustered indexes (this is political)

Tools are key.

Whether you use monitoring tools or DMV scripts:

- Know how to use the tool. Spend time reading the manual and experimenting with options.
- Practice, practice, practice. Don't take things for granted – run experiments to test yourself.
- Measure before & after, and prove that your change made a difference.

FirstResponderKit.org

sp_BlitzIndex

- Index health in this one database
- Add @Mode = 4 for smaller issues

sp_BlitzCache

- Most resource-intensive plans in the cache
- @SortOrder = 'reads' shows which queries read the most data, probably need indexes

All open source, MIT License.

You can do this.

For questions, leave comments on the relevant module.

For private help after the class, email Help@BrentOzar.com with:

- A note that you were in this class
- sp_Blitz @CheckServerInfo = 1
- sp_BlitzFirst @SinceStartup= 1

Thanks, and I hope you had a great time!