

## Default Maintenance Plans

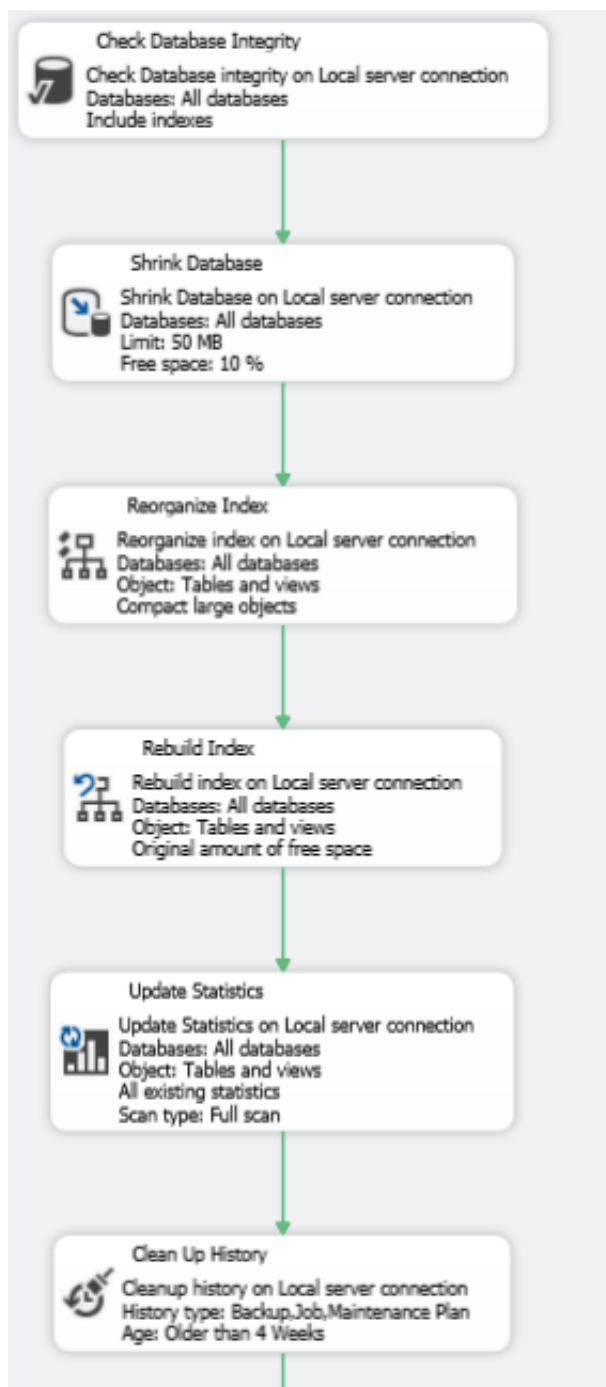
Shrinking a database can seem like a helpful way to save space and resources, particularly if a database has grown substantially over time. However, SQL Server's default maintenance plans that include database shrinking can lead to several performance issues. Here's why:

1. **Fragmentation:** When a database is shrunk, the data pages are reorganized to create a contiguous block of free space at the end of the data file, which can cause severe fragmentation. This fragmentation can significantly degrade the performance of your SQL Server because it has to jump around to different locations on disk to read a single object.
2. **Performance Impact:** The shrink operation can be resource-intensive and can negatively impact the overall performance of your SQL Server. If a shrink operation is executed during peak hours, it may cause slowness in the system and affect the user experience.
3. **File Growth:** Shrinking a database only to have it grow again is counterproductive. It can lead to file system fragmentation and reduces the performance of SQL Server. This constant growing and shrinking, also known as "yo-yoing", can cause inconsistency in database performance.
4. **Index Damage:** Every time you shrink the database, SQL Server might need to rearrange the pages, damaging the indexes. You'd need to rebuild the indexes after the shrink, which requires additional resources.
5. **Unused Space:** While shrinking a database does free up disk space, that space is often left unused. Unless you have an immediate need for that space, you're not gaining much by shrinking the database.

Instead of regularly shrinking your databases, a better approach to maintain the health of your SQL Server environment is to monitor and manage the growth of your databases, plan your storage requirements properly, and schedule regular maintenance tasks like updating statistics, checking database integrity, and optimizing or rebuilding indexes. It's recommended to only perform a SHRINK operation as a last resort or under careful consideration and supervision.

The problem with using the default maintenance plans is that SQL Server Management Studio makes it easy to turn on some features that are really bad practice.

You may end up with a maintenance plan that looks something like the following if you just take all the options.



There are several problems with this layout.

1. First and foremost, running Shrink Database is a really bad idea most of the time. It is especially bad to

run it on a daily or weekly basis. For more details see this blog post: [SteveStedman.com](http://SteveStedman.com) – [DBCC ShrinkDatabase – I want to shrink my database.](#)

There are very few occasions that will justify shrinking your database, and these should be few and far between. I have never come across a situation where setting the automatic shrink, or regularly scheduled shrinking of the database was a good idea. Don't shrink your database unless you have a really good understanding of the negative impacts, and you have a significant reason to do so. Often times if the reason to shrink your database is to regain disk space, it may be smarter to move some files around, or to add additional disks to get the space you need.

2. Reorganize Index followed by Rebuilding Indexes, then Updating statistics.

Why is this bad, well... when you rebuild your indexes it is similar to dropping the index and re-creating it, on enterprise edition it re-creates the index then switches it out. Either way it is throwing out the structure of the old index and rebuilding it, rebuilding an index also updates statistics with a full scan. So reorganizing the index is a complete waste before rebuilding the index because all the work done on the reorganize gets thrown out when it gets rebuilt. Next updating statistics is also a waste, since there is no need to update statistics after the index is rebuilt because the statistics would have been rebuilt also.

3. Check Database Integrity. This is a very resource intensive process to run, and typically you would want to run it on a different schedule than the other daily maintenance tasks. Because it typically takes a while to run you can run it in parallel with these other tasks rather than sequentially. You probably want to have different alerting enabled if the Check database integrity (checkdb) task fails because this is a critical issue that requires immediate attention.

#### Related Links

- [SteveStedman.com](http://SteveStedman.com) – [DBCC ShrinkDatabase – I want to shrink my database.](#)