

# Be Proactive: A Good DBA Goes Looking For Signs of Trouble

# BE PROACTIVE: A GOOD DBA GOES LOOKING FOR SIGNS OF TROUBLE

Database administrators (DBAs) are busier than ever these days, so fighting fires is the last thing they want to spend their time doing. A proactive approach to database performance and maintenance helps keep them ahead of trouble and prevent minor problems from turning into full-blown emergencies. This post discusses routine database maintenance and the process for setting up SQL Server Agent alerts for corruption, resource issues, and fatal errors. It also covers additional issues such as index analysis and slow queries that will help keep your databases running.

## ROUTINE DATABASE MAINTENANCE

DBAs know they need to perform backups, but they often cannot give this task as much attention as it deserves. In particular, they need to consider the implications of backups regarding recovery. Many DBAs feel their backup procedures are adequate because they take a full backup at regular intervals. However, it is important to remember that backup frequency determines the potential data loss resulting from a recovery. Such data loss comprises the data that we created after the backup, but before the recovery.

### BACKUPS

A recovery point object (RPO) describes the maximum amount of data an organization is willing to lose after a recovery. The RPO should dictate the backup schedule rather than the other way around, which is a trap many DBAs fall into. No executive wants to lose any data, but the cost of achieving zero data loss is prohibitively high in most cases. Defining the RPO is a matter of assessing the impact of data loss for specific time periods.

Another trap that DBAs often fall into is failing to back up the system databases and the user databases. During the total loss of the user database, a DBA would have to rebuild the database from scratch. The lack of system database backups means the new user database would not have any agent jobs, logins and permissions.

It is also important to ensure that backups are successful. DBAs should use the CHECKSUM option when creating a backup which performs some corruption checks. RESTORE VERIFYONLY is another option for verifying backups. This command ensures the backup is complete and readable, although it does not restore the database.

The backups that a DBA may perform include full, differential, log and full copy only backups. A full backup contains all the data in the database, while a differential backup contains only the data that has changed since the last full backup. Many DBAs believe they have to restore differential backups in chronological order to reach the desired recovery point, but this is not the case. It is only necessary to restore the most recent full backup before the recovery point, followed by the most recent differential backup.

SQL Server bases log backups on the latest full backup, but they also include all transactions that have occurred since the previous log backup. Unlike differential backups, log we must restore backups in order to reach the recovery point.

A full copy only backup is independent of the regular backup sequence that connects full, differential and log backups. The reason is that a full copy only backup does not change the log sequence number (LSN). This type of backup serves ad hoc purposes, such as making a copy of the database to restore on a development or testing environment.

## RECOVERY

If you do not test your backups, then you do not have backups, you just have files. Being prepared for a database failure means that you need to restore your backups. You need to know how long it takes to retrieve the backup files, which we can store locally or on a network server. It is also important to know the time needed to restore the backup, including the log files.

Senior executives will have an urgent desire to know when this process will be complete, making it important to provide an estimate that is as accurate as possible. In addition, you need to test your restore scripts beforehand, so they are ready for use. The last thing you want is to Google “How do I restore a transaction log?” while an executive is looking over your shoulder.

## INTEGRITY CHECKS

Database integrity checks are also a key part of database maintenance. DBCC CHECKDB is a full corruption check on the entire database, which some DBAs run before every backup. The primary reason for this practice is that some database servers like SQL Server allow you to back up a corrupt database, although it would not restore that backup. Therefore, it is important to ensure that we do not corrupt the database before backing it up.

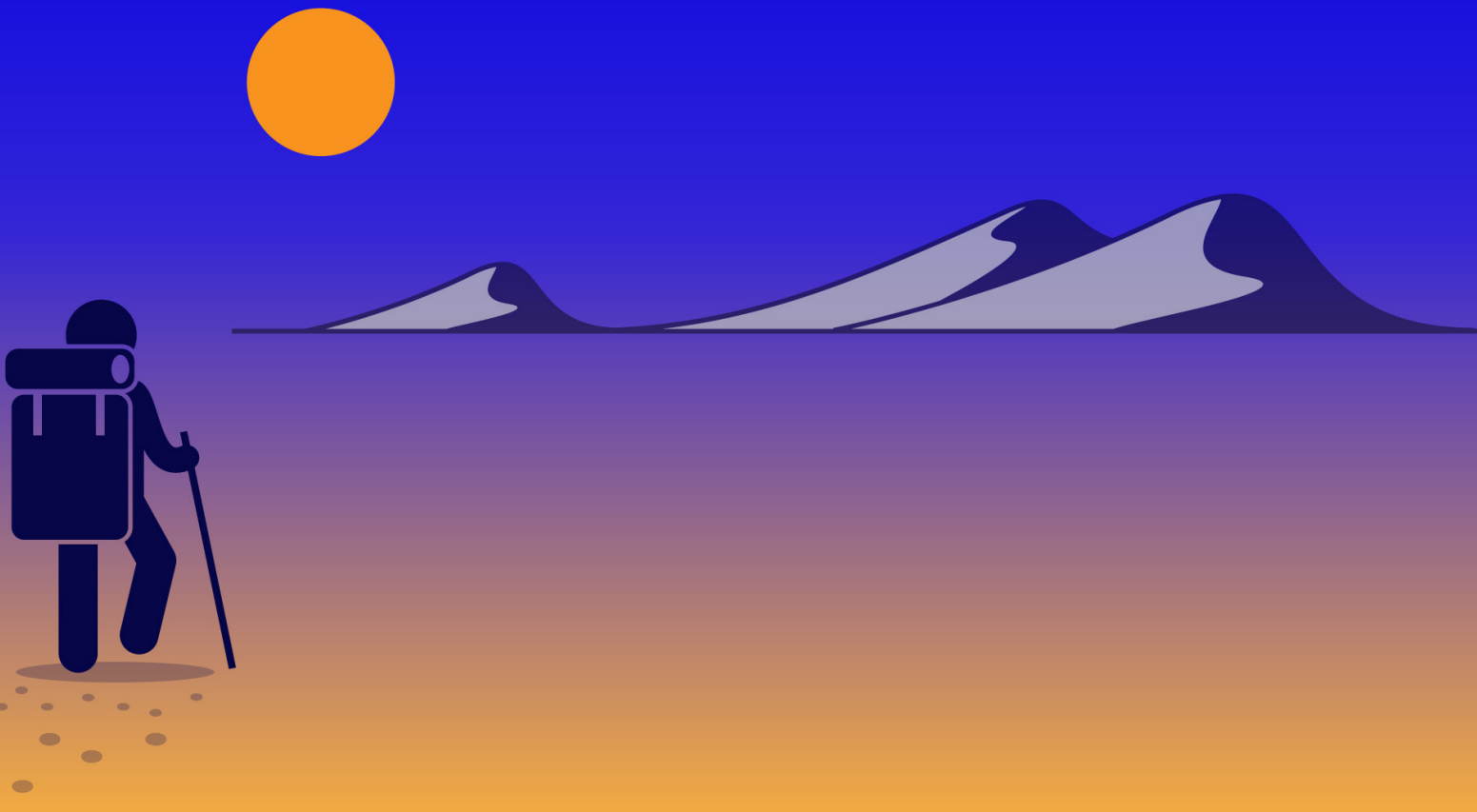
The biggest disadvantage of using CHECKDB is that it is resource intensive. It is not unusual for CHECKDB to take days to run on a multi-terabyte database, making it essential to schedule this process during a maintenance window. Some environments use a dedicated server that does nothing but run CHECKDB on databases, which is a recommended practice if a server is available for this purpose. As with backups, the frequency of integrity checks depends on the amount of data you can afford to lose. executive is looking over your shoulder.

## AGENT ALERTS

DBAs should also set up alerting to provide notification when database failures occur. Microsoft built this capability into the SQL Server agent, which will send a notification when it finds a qualifying condition. SQL Server errors with a severity between 17 and 25 inclusive are important, which include database engine errors and resource errors. SQL Server DBAs should also receive alerts on corruption errors 823, 824, 825, and 829.

## QUERY PERFORMANCE

“Proper indexing is one of the best performance enhancements you can make to your database”, according to Microsoft. However, most database developers lack a deep understanding of indexes. They use features like object-related modeling (ORM) and database engine tuning advisor (DTA) to create tables without knowing much about good database design. Issues to look for include indexes that are highly used, unused, fragmented or missing. DBAs should perform this process regularly to improve query performance.



# IDERA'S SOLUTION

## IDERA's SQL Doctor

IDERA's SQL Doctor helps database administrators to tune SQL Server performance, security, and disaster recovery via expert recommendations in physical, virtual, and cloud environments - including managed cloud databases. Unlike its competitors, it provides display health of all SQL Servers, generation of ready-to-run SQL scripts to optimize and undo optimization, limiting of analysis to specified databases, applications, and performance categories, and real-time, as-needed, and scheduled checkups.

Start for FREE

