

# Not Using All Cores

It is like the expression “not firing on all cylinders” or “a few cards short of a full deck”.

There are several reasons why SQL Server might not be using all available cores or might be running at a maximum of 50% CPU load. Here are some common scenarios and settings that can lead to such behavior:

## 1. Max Degree of Parallelism (MAXDOP):

- SQL Server uses the MAXDOP setting to control the number of processors used for the execution of a query in a parallel plan. If this is set too low, SQL Server might not utilize all available cores. It's important to set this appropriately based on your system's configuration and workload.

## 2. Resource Governor:

- The Resource Governor is a feature in SQL Server that allows you to manage SQL Server workloads and resources by specifying limits on resource consumption by incoming requests. If the Resource Governor is configured to limit CPU usage for certain workloads or users, it might prevent SQL Server from using full CPU capacity.

## 3. Affinity Mask:

- The affinity mask option binds SQL Server to a specified subset of CPUs. If this is configured incorrectly, it might restrict SQL Server to fewer cores than are available, leading to underutilization.

## 4. Virtualization:

- If SQL Server is running on a virtual machine, the virtual host configuration can limit the number of cores assigned to the SQL Server VM. Additionally, other VMs on the same host might be consuming resources, affecting the performance and resource availability for your SQL Server instance.

## 5. Licensing Limitations:

- Certain editions of SQL Server have core usage limitations. For example, SQL Server Standard Edition is limited in the number of cores it can use. Ensure your edition supports full utilization of your hardware.

**6. Non-parallel Queries:**

- Not all queries can be parallelized. Queries that are not parallelized due to their nature or due to settings (like MAXDOP 1) will not utilize all cores, as they are designed to run on a single thread.

**7. CPU Throttling:**

- On some systems, especially on virtual environments, CPU resources can be throttled based on the system policy or to prevent overheating. This can limit the amount of CPU available to SQL Server.

**8. Wait Statistics:**

- High waits or blocking can cause SQL Server to appear less busy than it is. This doesn't necessarily mean SQL Server isn't trying to use all the cores, but rather that something is preventing it from doing so.

When diagnosing why SQL Server isn't utilizing all cores or is capped at 50% CPU, it's crucial to look at the system as a whole, including SQL Server configuration, hardware, virtualization settings, and the nature of the workload. Tools like Database Health Monitor can be invaluable in this situation. It provides insights into server health, performance, and can help identify bottlenecks or misconfigurations affecting CPU usage. You can learn more and download it for free for single-server connections at [Database Health Monitor](#). If you're looking to improve your SQL Server performance and want a more in-depth understanding, consider enrolling in Stedman's SQL School classes at [Stedman.us/school](http://stedman.us/school).