



# Datenbankkonfiguration

## Enterprise applications

NetApp

February 10, 2026

This PDF was generated from <https://docs.netapp.com/de-de/ontap-apps-dbs/mssql/mssql-cpu-configuration.html> on February 10, 2026. Always check docs.netapp.com for the latest.

# Inhalt

- Datenbankkonfiguration ..... 1
  - CPU-Konfiguration ..... 1
    - Hyper-Threading ..... 1
    - Kerne und Lizenzierung ..... 1
    - CPU-Affinität ..... 2
    - Max. Parallelitätsgrad (MAXDOP) ..... 3
    - Max. Worker-Threads ..... 3
  - Speicherkonfiguration ..... 4
    - Max. Serverspeicher ..... 5
    - Uneinheitlicher Speicherzugriff ..... 6
    - Index Speicher erstellen ..... 6
    - Min. Arbeitsspeicher pro Abfrage ..... 6
  - Shared Instance oder dedizierte Instanz ..... 7
  - Tempdb-Dateien ..... 8

# Datenbankkonfiguration

## CPU-Konfiguration

Die SQL Server-Performance weist mehrere Abhängigkeiten von der CPU- und Kernkonfiguration auf.

### Hyper-Threading

Hyper-Threading bezieht sich entweder auf die Implementierung von simultanem Multithreading (SMT), wodurch die Parallelisierung von Berechnungen auf x86-Prozessoren verbessert wird. SMT ist sowohl für Intel als auch für AMD Prozessoren verfügbar.

Hyper-Threading führt zu logischen CPUs, die dem Betriebssystem als physische CPUs angezeigt werden. SQL Server erkennt dann die zusätzlichen CPUs und nutzt sie so, als gäbe es mehr Kerne als physisch vorhanden. Durch eine stärkere Parallelisierung kann die Performance erheblich gesteigert werden.

Der Nachteil hierbei ist, dass jede SQL Server-Version ihre eigenen Einschränkungen hinsichtlich der Rechenleistung hat, die sie verwenden kann. Weitere Informationen finden Sie unter ["Kapazitätsbeschränkungen für die Datenverarbeitung nach Edition von SQL Server"](#).

### Kerne und Lizenzierung

Es gibt zwei Optionen für die Lizenzierung von SQL Server. Die erste wird als Server + Client Access License (CAL)-Modell bezeichnet; die zweite ist das pro Prozessor-Core-Modell. Obwohl Sie mit der Server + CAL-Strategie auf alle in SQL Server verfügbaren Produktfunktionen zugreifen können, gibt es eine Hardwaregrenze von 20 CPU-Kernen pro Sockel. Selbst wenn Sie SQL Server Enterprise Edition + CAL für einen Server mit mehr als 20 CPU-Kernen pro Sockel verwenden, kann die Anwendung nicht alle diese Kerne gleichzeitig auf dieser Instanz verwenden.

Das Bild unten zeigt die SQL Server-Protokollmeldung nach dem Start, die die Durchsetzung des Core-Limits anzeigt.

```

2017-01-11 07:16:30.71 Server      Microsoft SQL Server 2016
(RTM) - 13.0.1601.5 (X64)
Apr 29 2016 23:23:58
Copyright (c) Microsoft Corporation
Enterprise Edition (64-bit) on Windows Server 2016
Datacenter 6.3 <X64> (Build 14393: )

2017-01-11 07:16:30.71 Server      UTC adjustment: -8:00
2017-01-11 07:16:30.71 Server      (c) Microsoft Corporation.
2017-01-11 07:16:30.71 Server      All rights reserved.
2017-01-11 07:16:30.71 Server      Server process ID is 10176.
2017-01-11 07:16:30.71 Server      System Manufacturer:
'FUJITSU', System Model: 'PRIMERGY RX2540 M1'.
2017-01-11 07:16:30.71 Server      Authentication mode is MIXED.
2017-01-11 07:16:30.71 Server      Logging SQL Server messages
in file 'C:\Program Files\Microsoft SQL Server
\MSSQL13.MSSQLSERVER\MSSQL\Log\ERRORLOG'.
2017-01-11 07:16:30.71 Server      The service account is 'SEA-
TM\FUJIA2R30$'. This is an informational message; no user action
is required.
2017-01-11 07:16:30.71 Server      Registry startup parameters:
-d C:\Program Files\Microsoft SQL Server
\MSSQL13.MSSQLSERVER\MSSQL\DATA\master.mdf
-e C:\Program Files\Microsoft SQL Server
\MSSQL13.MSSQLSERVER\MSSQL\Log\ERRORLOG
-l C:\Program Files\Microsoft SQL Server
\MSSQL13.MSSQLSERVER\MSSQL\DATA\mastlog.ldf
-T 3502
-T 834
2017-01-11 07:16:30.71 Server      Command Line Startup
Parameters:
-a "MSSQLSERVER"
2017-01-11 07:16:30.72 Server      SQL Server detected 2 sockets
with 18 cores per socket and 36 logical processors per socket,
72 total logical processors; using 40 logical processors based
on SQL Server licensing. This is an informational message; no
user action is required.
2017-01-11 07:16:30.72 Server      SQL Server is starting at

```

Um alle CPUs zu verwenden, sollten Sie daher die Prozessorkern-Lizenz verwenden. Weitere Informationen zur SQL Server-Lizenzierung finden Sie unter ["SQL Server 2022: Ihre moderne Datenplattform"](#).

## CPU-Affinität

Es ist unwahrscheinlich, dass Sie die Standardeinstellungen für die Prozessoraffinität ändern müssen, es sei denn, Sie stoßen auf Leistungsprobleme, aber es lohnt sich immer noch zu verstehen, was sie sind und wie sie funktionieren.

SQL Server unterstützt die Prozessoraffinität durch zwei Optionen:

- CPU-Affinitätsmaske
- Affinity-E/A-Maske

SQL Server verwendet alle CPUs, die über das Betriebssystem verfügbar sind (wenn die Prozessorkern-Lizenz gewählt wird). Es erstellt auch Scheduler für jede CPU, um die Ressourcen für jeden beliebigen Workload optimal zu nutzen. Beim Multitasking kann das Betriebssystem oder andere Anwendungen auf dem Server die Prozess-Threads von einem Prozessor zum anderen wechseln. SQL Server ist eine ressourcenintensive Applikation und die Performance kann in diesem Fall beeinträchtigt werden. Um die Auswirkungen zu minimieren, können Sie die Prozessoren so konfigurieren, dass die gesamte SQL Server-Last an eine vorgewählte Prozessorgruppe weitergeleitet wird. Dies wird durch die CPU Affinitätsmaske erreicht.

Die Affinity I/O-Maskenoption bindet SQL Server-Festplatten-I/O an eine Teilmenge von CPUs. In SQL Server-OLTP-Umgebungen kann diese Erweiterung die Performance von SQL Server-Threads, die I/O-Vorgänge ausgeben, erheblich verbessern.

## Max. Parallelitätsgrad (MAXDOP)

Standardmäßig verwendet SQL Server während der Abfrageausführung alle verfügbaren CPUs, wenn die Prozessorkern-Lizenz ausgewählt wurde.

Dies ist zwar hilfreich bei umfangreichen Abfragen, kann jedoch zu Leistungsproblemen und zur Begrenzung der Parallelität führen. Ein besserer Ansatz besteht darin, die Parallelität auf die Anzahl der physischen Kerne in einem einzelnen CPU-Socket zu beschränken. Beispielsweise sollte auf einem Server mit zwei physischen CPU-Sockeln mit 12 Kernen pro Socket, unabhängig von Hyper-Threading, MAXDOP auf 12 gesetzt werden. MAXDOP Die zu verwendende CPU kann nicht eingeschränkt oder diktiert werden. Stattdessen beschränkt es die Anzahl der CPUs, die von einer einzelnen Batch-Abfrage verwendet werden können.



**NetApp empfiehlt** für DSS wie Data Warehouses, beginnen Sie mit MAXDOP 50 und erkunden Sie Tuning auf oder ab, falls erforderlich. Stellen Sie sicher, dass Sie die kritischen Abfragen in Ihrer Anwendung messen, wenn Sie Änderungen vornehmen.

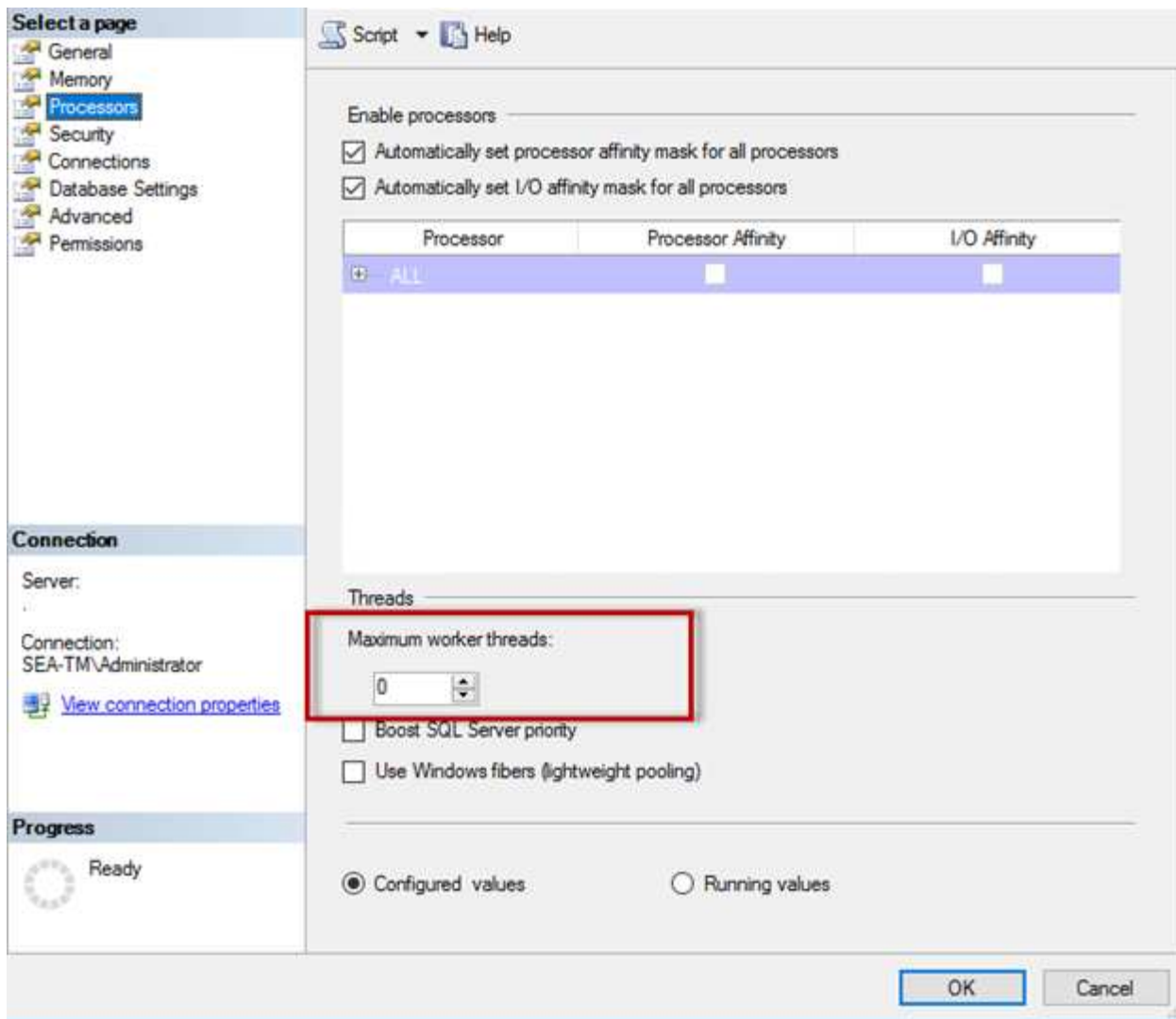
## Max. Worker-Threads

Die Option Max. Worker-Threads hilft, die Leistung zu optimieren, wenn eine große Anzahl von Clients mit SQL Server verbunden ist.

Normalerweise wird für jede Abfrage ein separater Betriebssystem-Thread erstellt. Wenn Hunderte von gleichzeitigen Verbindungen zu SQL Server hergestellt werden, kann eine Konfiguration mit einem Thread pro Abfrage übermäßige Systemressourcen beanspruchen. Die `max worker threads` Option verbessert die Leistung, indem SQL Server in die Lage versetzt wird, einen Pool von Worker-Threads zu erstellen, die gemeinsam eine größere Anzahl von Abfrage-Anforderungen verarbeiten können.

Der Standardwert ist 0, wodurch SQL Server die Anzahl der Worker-Threads beim Start automatisch konfigurieren kann. Dies funktioniert für die meisten Systeme. Max Worker-Threads sind eine erweiterte Option und sollten nicht ohne Unterstützung durch einen erfahrenen Datenbankadministrator (DBA) geändert werden.

Wann sollten Sie SQL Server so konfigurieren, dass mehr Worker-Threads verwendet werden? Wenn die durchschnittliche Länge der Arbeitswarteschlange für jeden Scheduler über 1 liegt, können Sie vom Hinzufügen weiterer Threads zum System profitieren, jedoch nur, wenn die Last nicht CPU-gebunden ist oder andere schwere Wartezeiten auftritt. Wenn einer dieser Vorgänge stattfindet, sind weitere Threads nicht hilfreich, da sie schließlich auf andere Systemengpässe warten müssen. Weitere Informationen zu `max worker threads` finden Sie unter ["Konfigurieren Sie die Option max Worker Threads Server Configuration"](#).



## Konfigurieren von max Worker-Threads mit SQL Server Management Studio

Das folgende Beispiel zeigt, wie Sie die Option Max. Work Threads mit T-SQL konfigurieren.

```
EXEC sp_configure 'show advanced options', 1;
GO
RECONFIGURE ;
GO
EXEC sp_configure 'max worker threads', 900 ;
GO
RECONFIGURE;
GO
```

## Speicherkonfiguration

Im folgenden Abschnitt werden die SQL Server-Speichereinstellungen erläutert, die zur Optimierung der Datenbankleistung erforderlich sind.

## Max. Serverspeicher

Mit der Option „Max. Serverspeicher“ wird die maximale Speichergröße festgelegt, die die SQL Server-Instanz verwenden kann. Sie wird in der Regel verwendet, wenn mehrere Anwendungen auf demselben Server ausgeführt werden, auf dem SQL Server ausgeführt wird, und Sie sicherstellen möchten, dass diese Anwendungen über genügend Arbeitsspeicher verfügen, um ordnungsgemäß zu funktionieren.

Einige Anwendungen verwenden nur den verfügbaren Speicher, wenn sie starten, und fordern keinen zusätzlichen Speicher an, selbst wenn sie unter Speicherdruck stehen. Hier kommt die maximale Serverspeichereinstellung ins Spiel.

Auf einem SQL Server-Cluster mit mehreren SQL Server-Instanzen könnte jede Instanz mit Ressourcen konkurrieren. Durch die Festlegung einer Speichergrenze für jede SQL Server-Instanz kann eine optimale Performance für jede Instanz gewährleistet werden.



**NetApp empfiehlt**, mindestens 4 GB bis 6 GB RAM für das Betriebssystem zu belassen, um Leistungsprobleme zu vermeiden.

The screenshot shows the 'Server Memory' configuration window in SQL Server Enterprise Manager. The left pane has 'Memory' selected under 'Select a page'. The 'Server memory options' section is highlighted with a red box, showing 'Minimum server memory (in MB)' set to 0 and 'Maximum server memory (in MB)' set to 120832. Below this, the 'Other memory options' section shows 'Index creation memory (in KB)' set to 0 and 'Minimum memory per query (in KB)' set to 1024. At the bottom, the 'Configured values' radio button is selected, and the 'Progress' bar shows 'Ready'. The 'OK' and 'Cancel' buttons are at the bottom right.

### Anpassen des minimalen und maximalen Serverspeichers mit SQL Server Management Studio

Die Verwendung von SQL Server Management Studio zur Anpassung des minimalen oder maximalen Serverspeichers erfordert einen Neustart des SQL Server-Dienstes. Sie können den Serverspeicher auch

mithilfe von Transact SQL (T-SQL) anpassen, indem Sie diesen Code verwenden:

```
EXECUTE sp_configure 'show advanced options', 1
GO
EXECUTE sp_configure 'min server memory (MB)', 2048
GO
EXEC sp_configure 'max server memory (MB)', 120832
GO
RECONFIGURE WITH OVERRIDE
```

## Uneinheitlicher Speicherzugriff

NUMA (Nonuniform Memory Access) ist eine Technologie zur Optimierung des Speicherzugriffs, die dazu beiträgt, eine zusätzliche Belastung des Prozessorbusses zu vermeiden.

Wenn NUMA auf einem Server konfiguriert ist, auf dem SQL Server installiert ist, ist keine zusätzliche Konfiguration erforderlich, da SQL Server NUMA-fähig ist und auf NUMA-Hardware eine gute Performance erzielt.

## Index Speicher erstellen

Die Option Index create Memory ist eine weitere erweiterte Option, die normalerweise nicht von den Standardwerten geändert werden muss.

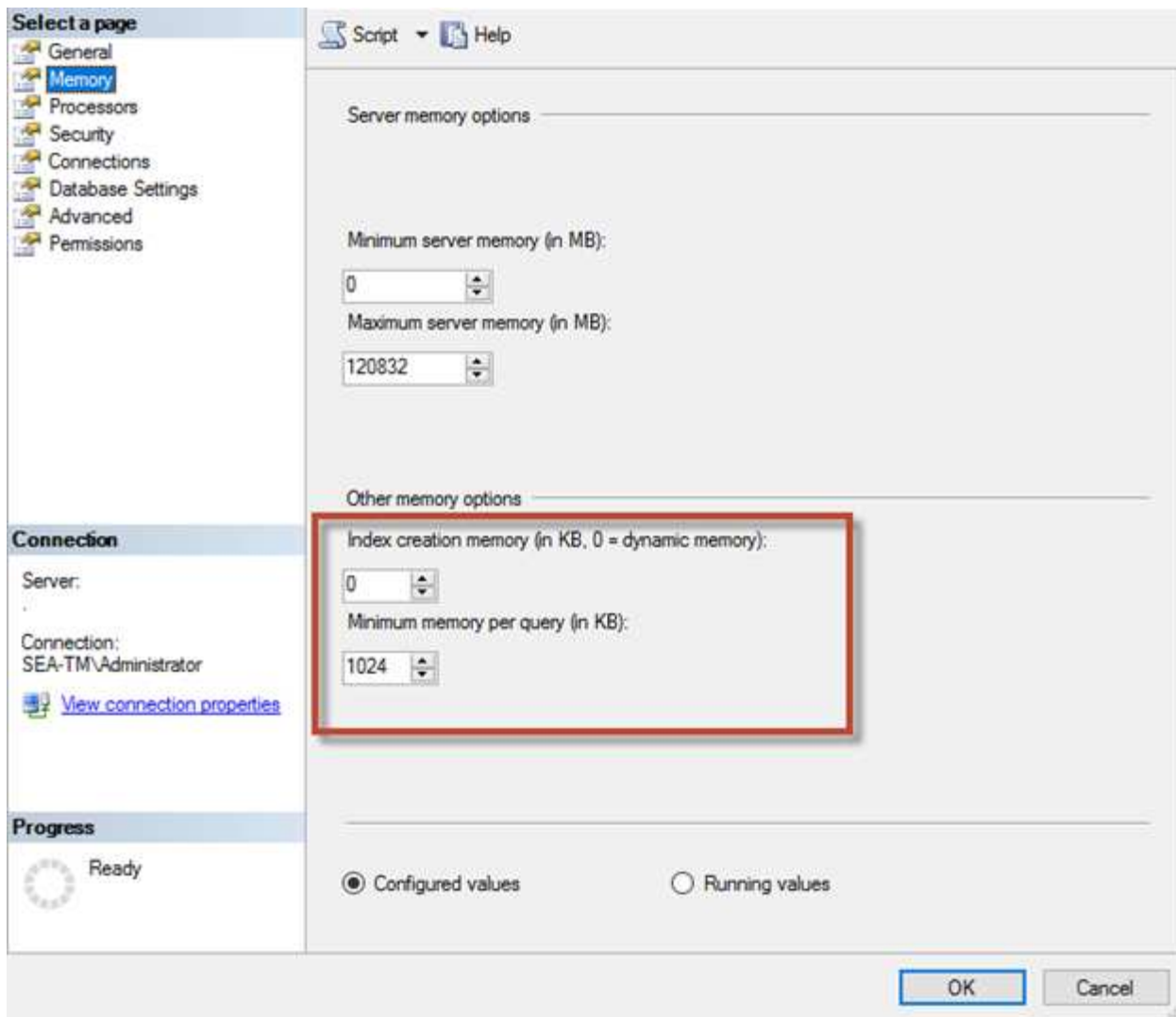
Er steuert die maximale RAM-Größe, die ursprünglich für die Erstellung von Indizes zugewiesen wurde. Der Standardwert für diese Option ist 0, was bedeutet, dass sie von SQL Server automatisch verwaltet wird. Wenn Sie jedoch Schwierigkeiten beim Erstellen von Indizes haben, sollten Sie den Wert dieser Option erhöhen.

## Min. Arbeitsspeicher pro Abfrage

Wenn eine Abfrage ausgeführt wird, versucht SQL Server, die optimale Speichergröße für eine effiziente Ausführung zuzuweisen.

Standardmäßig weist die Einstellung Min. Speicher pro Abfrage >= bis 1024 KB für jede auszufüllende Abfrage zu. Es empfiehlt sich, diese Einstellung auf den Standardwert zu belassen, damit SQL Server die für Indexerstellung zugewiesene Speichermenge dynamisch verwalten kann. Wenn SQL Server jedoch über mehr RAM verfügt, als für eine effiziente Ausführung erforderlich ist, kann die Leistung einiger Abfragen erhöht werden, wenn Sie diese Einstellung erhöhen. Solange also auf dem Server, der nicht von SQL Server verwendet wird, Speicher verfügbar ist, können andere Anwendungen oder das Betriebssystem diese Einstellung erhöhen, was die gesamte SQL Server-Leistung verbessern kann. Wenn kein freier Speicher verfügbar ist, kann eine Erhöhung dieser Einstellung die Gesamtleistung beeinträchtigen.





## Shared Instance oder dedizierte Instanz

SQL Server kann als einzelne Instanz pro Server oder als mehrere Instanzen konfiguriert werden. Die richtige Entscheidung hängt in der Regel von Faktoren ab, z. B. ob der Server für die Produktion oder Entwicklung verwendet werden soll, ob die Instanz für den Geschäftsbetrieb und die Leistungsziele entscheidend ist.

Konfigurationen für gemeinsam genutzte Instanzen sind zwar zunächst einfacher zu konfigurieren, können jedoch zu Problemen führen, bei denen Ressourcen aufgeteilt oder gesperrt werden. Dies führt wiederum zu Leistungsproblemen für andere Anwendungen, bei denen Datenbanken auf der gemeinsam genutzten SQL Server-Instanz gehostet werden.

Die Fehlerbehebung bei Performance-Problemen kann kompliziert sein, da Sie herausfinden müssen, welche Instanz die eigentliche Ursache ist. Diese Frage wird gegen die Kosten von Betriebssystemlizenzen und SQL Server-Lizenzen abgewogen. Wenn die Applikations-Performance oberste Priorität hat, ist eine dedizierte Instanz sehr empfehlenswert.

Microsoft lizenziert SQL Server pro Kern auf Serverebene und nicht pro Instanz. Aus diesem Grund sind Datenbankadministratoren versucht, so viele SQL Server-Instanzen zu installieren, wie der Server verarbeiten kann, um Lizenzierungskosten zu sparen, was später zu größeren Performanceproblemen führen kann.



**NetApp empfiehlt**, wenn möglich dedizierte SQL Server-Instanzen zu wählen, um optimale Leistung zu erzielen.

## Tempdb-Dateien

Die Tempdb-Datenbank kann stark genutzt werden. Neben der optimalen Platzierung von Benutzerdatenbankdateien auf ONTAP ist die Platzierung von tempdb-Datendateien auch wichtig, um die Zuweisungskonflikte zu verringern. Tempdb sollte auf einer separaten Festplatte abgelegt und nicht für Benutzerdatendateien freigegeben werden.

Seitenkonflikte können auf den Seiten Global Allocation Map (GAM), Shared Global Allocation Map (SGAM) oder Page Free Space (PFS) auftreten, wenn SQL Server auf spezielle Systemseiten schreiben muss, um neue Objekte zuzuweisen. Verriegelungen sperren diese Seiten im Speicher. In einer stark ausgelasteten SQL Server-Instanz kann es lange dauern, bis ein Latch auf einer Systemseite in tempdb abgerufen wird. Dies führt zu längeren Abfragezeiten und wird als Latch Contention bezeichnet. Lesen Sie die folgenden Best Practices für das Erstellen von tempdb-Datendateien:

- Für  $\leq$  bis 8 Kerne: Tempdb-Datendateien = Anzahl der Kerne
- Für  $> 8$  Kerne: 8 tempdb-Datendateien
- Die tempdb-Datendatei sollte mit gleicher Größe erstellt werden

Mit dem folgenden Beispielskript wird tempdb geändert, indem acht tempdb-Dateien gleicher Größe erstellt und tempdb auf den Mount-Punkt für SQL Server 2012 und höher verschoben `C:\MSSQL\tempdb` wird.

```
use master

go

-- Change logical tempdb file name first since SQL Server shipped with
logical file name called tempdev

alter database tempdb modify file (name = 'tempdev', newname =
'tempdev01');

-- Change location of tempdev01 and log file

alter database tempdb modify file (name = 'tempdev01', filename =
'C:\MSSQL\tempdb\tempdev01.mdf');

alter database tempdb modify file (name = 'templog', filename =
'C:\MSSQL\tempdb\templog.ldf');

GO

-- Assign proper size for tempdev01
```

```

ALTER DATABASE [tempdb] MODIFY FILE ( NAME = N'tempdev01', SIZE = 10GB );

ALTER DATABASE [tempdb] MODIFY FILE ( NAME = N'templog', SIZE = 10GB );

GO

-- Add more tempdb files

ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev02', FILENAME =
N'C:\MSSQL\tempdb\tempdev02.ndf' , SIZE = 10GB , FILEGROWTH = 10%);

ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev03', FILENAME =
N'C:\MSSQL\tempdb\tempdev03.ndf' , SIZE = 10GB , FILEGROWTH = 10%);

ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev04', FILENAME =
N'C:\MSSQL\tempdb\tempdev04.ndf' , SIZE = 10GB , FILEGROWTH = 10%);

ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev05', FILENAME =
N'C:\MSSQL\tempdb\tempdev05.ndf' , SIZE = 10GB , FILEGROWTH = 10%);

ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev06', FILENAME =
N'C:\MSSQL\tempdb\tempdev06.ndf' , SIZE = 10GB , FILEGROWTH = 10%);

ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev07', FILENAME =
N'C:\MSSQL\tempdb\tempdev07.ndf' , SIZE = 10GB , FILEGROWTH = 10%);

ALTER DATABASE [tempdb] ADD FILE ( NAME = N'tempdev08', FILENAME =
N'C:\MSSQL\tempdb\tempdev08.ndf' , SIZE = 10GB , FILEGROWTH = 10%);

GO

```

Ab SQL Server 2016 wird die Anzahl der für das Betriebssystem sichtbaren CPU-Kerne während der Installation automatisch erkannt. Auf Basis dieser Anzahl berechnet und konfiguriert SQL Server die Anzahl der für eine optimale Performance erforderlichen tempdb-Dateien.

## Copyright-Informationen

Copyright © 2026 NetApp. Alle Rechte vorbehalten. Gedruckt in den USA. Dieses urheberrechtlich geschützte Dokument darf ohne die vorherige schriftliche Genehmigung des Urheberrechtsinhabers in keiner Form und durch keine Mittel – weder grafische noch elektronische oder mechanische, einschließlich Fotokopieren, Aufnehmen oder Speichern in einem elektronischen Abrufsystem – auch nicht in Teilen, vervielfältigt werden.

Software, die von urheberrechtlich geschütztem NetApp Material abgeleitet wird, unterliegt der folgenden Lizenz und dem folgenden Haftungsausschluss:

DIE VORLIEGENDE SOFTWARE WIRD IN DER VORLIEGENDEN FORM VON NETAPP ZUR VERFÜGUNG GESTELLT, D. H. OHNE JEGLICHE EXPLIZITE ODER IMPLIZITE GEWÄHRLEISTUNG, EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE STILLSCHWEIGENDE GEWÄHRLEISTUNG DER MARKTGÄNGIGKEIT UND EIGNUNG FÜR EINEN BESTIMMTEN ZWECK, DIE HIERMIT AUSGESCHLOSSEN WERDEN. NETAPP ÜBERNIMMT KEINERLEI HAFTUNG FÜR DIREKTE, INDIREKTE, ZUFÄLLIGE, BESONDERE, BEISPIELHAFTE SCHÄDEN ODER FOLGESCHÄDEN (EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE BESCHAFFUNG VON ERSATZWAREN ODER -DIENSTLEISTUNGEN, NUTZUNGS-, DATEN- ODER GEWINNVERLUSTE ODER UNTERBRECHUNG DES GESCHÄFTSBETRIEBS), UNABHÄNGIG DAVON, WIE SIE VERURSACHT WURDEN UND AUF WELCHER HAFTUNGSTHEORIE SIE BERUHEN, OB AUS VERTRAGLICH FESTGELEGTER HAFTUNG, VERSCHULDENSUNABHÄNGIGER HAFTUNG ODER DELIKTSHAFTUNG (EINSCHLIESSLICH FAHRLÄSSIGKEIT ODER AUF ANDEREM WEGE), DIE IN IRGEND EINER WEISE AUS DER NUTZUNG DIESER SOFTWARE RESULTIEREN, SELBST WENN AUF DIE MÖGLICHKEIT DERARTIGER SCHÄDEN HINGEWIESEN WURDE.

NetApp behält sich das Recht vor, die hierin beschriebenen Produkte jederzeit und ohne Vorankündigung zu ändern. NetApp übernimmt keine Verantwortung oder Haftung, die sich aus der Verwendung der hier beschriebenen Produkte ergibt, es sei denn, NetApp hat dem ausdrücklich in schriftlicher Form zugestimmt. Die Verwendung oder der Erwerb dieses Produkts stellt keine Lizenzierung im Rahmen eines Patentrechts, Markenrechts oder eines anderen Rechts an geistigem Eigentum von NetApp dar.

Das in diesem Dokument beschriebene Produkt kann durch ein oder mehrere US-amerikanische Patente, ausländische Patente oder anhängige Patentanmeldungen geschützt sein.

ERLÄUTERUNG ZU „RESTRICTED RIGHTS“: Nutzung, Vervielfältigung oder Offenlegung durch die US-Regierung unterliegt den Einschränkungen gemäß Unterabschnitt (b)(3) der Klausel „Rights in Technical Data – Noncommercial Items“ in DFARS 252.227-7013 (Februar 2014) und FAR 52.227-19 (Dezember 2007).

Die hierin enthaltenen Daten beziehen sich auf ein kommerzielles Produkt und/oder einen kommerziellen Service (wie in FAR 2.101 definiert) und sind Eigentum von NetApp, Inc. Alle technischen Daten und die Computersoftware von NetApp, die unter diesem Vertrag bereitgestellt werden, sind gewerblicher Natur und wurden ausschließlich unter Verwendung privater Mittel entwickelt. Die US-Regierung besitzt eine nicht ausschließliche, nicht übertragbare, nicht unterlizenzierbare, weltweite, limitierte unwiderrufliche Lizenz zur Nutzung der Daten nur in Verbindung mit und zur Unterstützung des Vertrags der US-Regierung, unter dem die Daten bereitgestellt wurden. Sofern in den vorliegenden Bedingungen nicht anders angegeben, dürfen die Daten ohne vorherige schriftliche Genehmigung von NetApp, Inc. nicht verwendet, offengelegt, vervielfältigt, geändert, aufgeführt oder angezeigt werden. Die Lizenzrechte der US-Regierung für das US-Verteidigungsministerium sind auf die in DFARS-Klausel 252.227-7015(b) (Februar 2014) genannten Rechte beschränkt.

## Markeninformationen

NETAPP, das NETAPP Logo und die unter <http://www.netapp.com/TM> aufgeführten Marken sind Marken von NetApp, Inc. Andere Firmen und Produktnamen können Marken der jeweiligen Eigentümer sein.