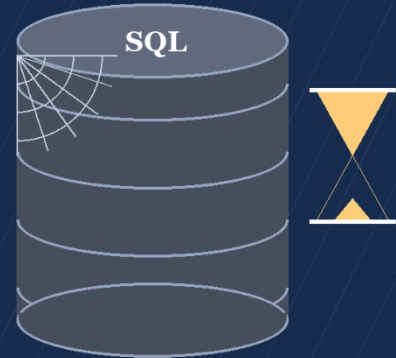


## SQL Server unter Klinikanwendungen: Warum Subsysteme langsam werden und was wirklich hilft

Microsoft 365 im Krankenhaus · SQL Server

### SQL Server unter Klinikanwendungen: Warum Subsysteme langsam werden und was wirklich hilft

*Der Hersteller betreut die Anwendung, die IT den Server. Die Datenbank dazwischen betreut die Sanduhr.*



boddenberg.de

#### WISSEN

Alle Beiträge der Serie zu Microsoft 365 und Klinik-IT an einem Ort.

› [Microsoft 365 im Krankenhaus](#)

#### BERATUNG

SQL-Performance-Analyse für Subsysteme und Fachanwendungen: messen, Ursachen benennen, mit dem Hersteller abstimmen.

› [Beratung für Krankenhäuser](#)

#### SCHULUNG

Für IT-Administratoren: SQL Server im Klinikbetrieb sicher warten, überwachen und dokumentieren.

› [Schulungen für Krankenhäuser](#)

Es gibt im Krankenhaus einen Satz, der zuverlässiger fällt, als der Kaffeeautomat im Arztzimmer funktioniert: „Das System ist heute wieder so langsam.“ Gemeint ist selten das KIS selbst, sondern eines der vielen Subsysteme und Fachanwendungen drumherum. Das LIS, das RIS, die Software der Medizintechnik, das CAFM, die Dienstplanung, die Küche, die Apotheke. Fast alle haben eines gemeinsam: Unter der Haube arbeitet ein Microsoft SQL Server, und um diesen SQL Server kümmert sich niemand so richtig.

Das ist kein Vorwurf, sondern Arbeitsteilung, die an genau einer Stelle nicht aufgeht. Der Hersteller verantwortet die Anwendung. Die IT verantwortet Server, Hypervisor, Sicherung und Netz. Die Datenbankinstanz liegt dazwischen, und dazwischen veraltet alles in aller Ruhe: Statistiken, Wartungsjobs, Konfiguration, Patchstand. Irgendwann wird neue Hardware bestellt, die Anwendung wird kurz schneller oder auch nicht, und nach einem halben Jahr ruft die Station wieder an.

Dieser Beitrag beschreibt, warum Subsysteme auf SQL Server langsam werden, welche Befunde in der Praxis immer wieder auftauchen und was tatsächlich hilft, wenn es um **SQL Server Performance im Krankenhaus** geht. Er gehört zur Serie [Microsoft 365 im Krankenhaus](#), auch wenn hier kein einziger Teams-Kanal vorkommt. Die Grundhaltung der Serie lautet: Patientendaten gehören ins KIS und seine Subsysteme. Genau deshalb verdienen diese Systeme eine Datenbank, die jemand pflegt.

#### FAKTEN · Das Wichtigste in Kürze

Viele Subsysteme laufen auf SQL Server, ohne dass jemand die Instanz selbst betreut. Typische

Folgen: gestoppter SQL Agent, keine Konsistenzprüfung, alte Statistiken, zu wenig Arbeitsspeicher.

Die Standard Edition nutzt laut Microsoft höchstens den kleineren Wert aus 4 Sockeln oder 24 Kernen (bis SQL Server 2022) bzw. 32 Kernen (SQL Server 2025). Eine falsch zugeschnittene VM verschenkt so einen Großteil ihrer vCPU.

Mehr Hardware hilft nur dem Teil der Antwortzeit, der an Hardware hängt. Abfragen, die riesige Datenmengen lesen, bleiben auch auf neuen Servern teuer.

Wirksam ist ein Zyklus aus Messung, Maßnahme und Nachmessung, abgestimmt mit dem Hersteller: Freigaben für Kompatibilitätsgrad und eigene Indizes, dazu ein geregelter Umgang mit eigenen Indizes vor jedem Update.

### **KOMPETENZBEREICH · SQL Server bei boddenberg.de**

Analyse, Härtung und Betrieb von SQL Server, von der Instanzkonfiguration über Wartung und Abfrageoptimierung bis zur Hochverfügbarkeit, sind in einem eigenen Bereich gebündelt:

[Kompetenzbereich SQL Server](#). Dort finden Sie die technischen Hintergründe, die dieser Beitrag nur anreißen kann.

## **Das Niemandsland zwischen Hersteller und IT**

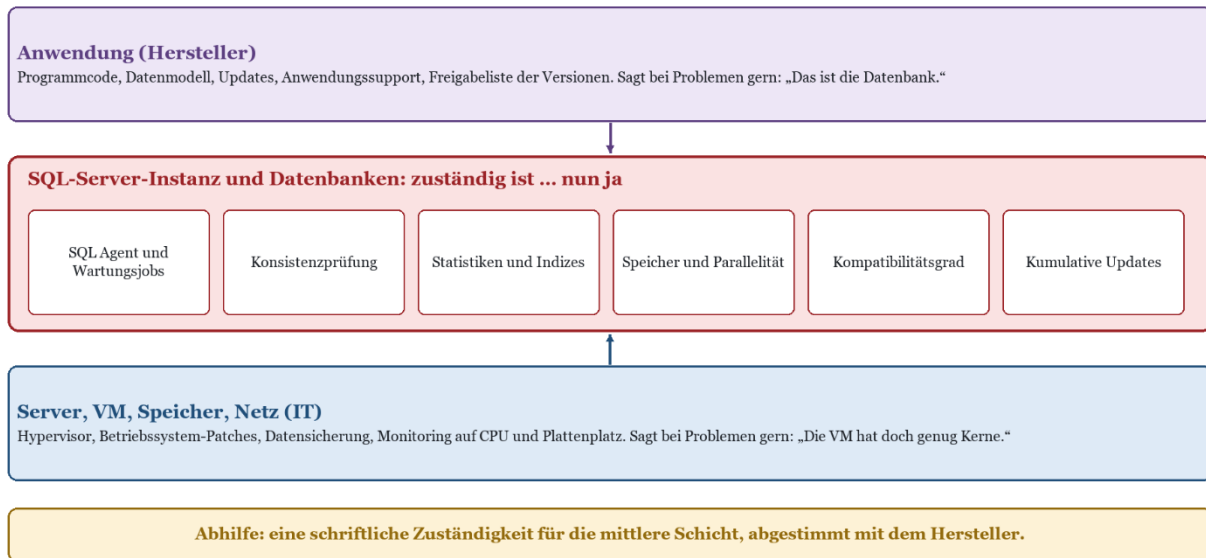
### **Wer betreut den SQL Server eigentlich?**

Fragen Sie in einem beliebigen Haus, wer für den SQL Server des Laborsystems zuständig ist. Sie bekommen mit großer Wahrscheinlichkeit zwei Antworten, die sich gegenseitig ausschließen. Die IT sagt: „Der Hersteller, der hat ihn installiert und hat ja auch Fernwartung.“ Der Hersteller sagt: „Wir betreuen die Anwendung. Die Infrastruktur stellt das Haus.“ Beide haben nach ihrem Vertrag recht, und genau das ist das Problem.

Das Ergebnis ist eine Instanz, die irgendwann ein Techniker des Herstellers nach Installationsanleitung aufgesetzt hat, meist unter Zeitdruck kurz vor dem Produktivstart. Danach hat sie niemand mehr angefasst. Die IT patcht das Betriebssystem, sichert die VM und überwacht den Plattenplatz. Der Hersteller spielt Updates der Anwendung ein. Was in der Instanz selbst passiert, schaut sich keiner an, solange niemand anruft. Und angerufen wird erst, wenn es auf Station schon niemand mehr lustig findet.

## Skizze 1: Das Niemandsland zwischen Hersteller und IT

*Jede Seite betreut ihre Schicht sauber. Die Datenbank fällt durch die Fuge.*



*Schematische Darstellung, keine bestimmte Anwendung.*

## Wie Subsysteme schleichend langsam werden

Langsamkeit kommt selten mit einem Knall. Sie kommt in Raten. Jedes Jahr kommen Millionen Zeilen dazu, weil im Krankenhaus nichts gelöscht wird, was irgendwann ein Prüfer sehen wollen könnte. Abfragen, die beim Produktivstart über eine kleine Tabelle liefen, laufen jetzt über eine große. Die Statistiken, auf deren Grundlage SQL Server seine Ausführungspläne wählt, beschreiben noch die Datenlage von damals. Der Arbeitsspeicher, der früher die ganze Datenbank fasste, fasst heute einen Bruchteil davon. Niemand merkt den einzelnen Schritt, aber in Summe wird aus der Sekunde eine halbe Minute.

Dazu kommt das typische Lastprofil eines Krankenhauses. Zur Visite, beim Schichtwechsel und kurz vor Feierabend in der Verwaltung greifen viele gleichzeitig zu. Nachts laufen Schnittstellen, Exporte und die Datensicherung. Wer die Performance nur misst, wenn gerade jemand anruft, misst meistens das Falsche. Und wer nachts misst, wundert sich, warum alles in Ordnung aussieht.

### **WARNUNG · Ohne Zuständigkeit keine Wartung**

Eine Datenbank ohne Wartung läuft nicht einfach weiter wie bisher. Sie wird langsamer, und Beschädigungen bleiben unentdeckt, bis jemand eine Sicherung zurückspielen will. Bei einem Subsystem mit Patientenbezug ist das kein Schönheitsfehler, sondern ein Verfügbarkeitsrisiko. Klären Sie die Zuständigkeit für die Instanz schriftlich, bevor Sie über Hardware reden.

## Die typischen Befunde: Was sich nach Jahren in der Instanz findet

Die folgenden Befunde tauchen so regelmäßig auf, dass man sie als Checkliste laminieren könnte. Keiner davon ist exotisch. Alle lassen sich mit Bordmitteln in wenigen Stunden feststellen, sofern jemand mit den nötigen Rechten hinschaut.

### **SQL Agent gestoppt, Wartungspläne, die nie liefen**

Der SQL Server Agent ist der Dienst, der geplante Aufgaben ausführt: Indexpflege, Aktualisierung der Statistiken, Konsistenzprüfung, oft auch die Sicherung des Transaktionsprotokolls. Steht der Dienst auf „Manuell“ oder ist er nach einem Neustart nie wieder angelaufen, passiert nichts davon.

Die Jobs stehen ordentlich angelegt in der Liste, der Verlauf ist leer. Ein Klassiker, der besonders gern nach dem Umzug auf eine neue VM auftritt.

Eine Sonderform: Die Anwendung läuft auf der Express Edition. Dort gibt es laut Microsoft gar keinen SQL Server Agent, dafür eine Obergrenze für die Datenbankgröße und einen stark begrenzten Puffercache. Für eine kleine Abteilungsanwendung ist das legitim. Für ein Subsystem, das seit Jahren wächst, ist es eine Zeitbombe mit gut lesbarer Anzeige.

### **Nie eine Konsistenzprüfung**

DBCC CHECKDB prüft die logische und physische Integrität einer Datenbank. Die Prüfung kostet Zeit und Plattenzugriffe, deshalb läuft sie idealerweise nachts oder am Wochenende. In vielen Instanzen ist sie nie gelaufen. Das fällt erst auf, wenn eine beschädigte Seite gelesen wird, und dann stellt sich die unangenehme Frage, seit wann die Beschädigung in allen Sicherungen mitreist. Die Antwort lautet gelegentlich: länger, als die Sicherungen aufbewahrt werden.

### **Uralte Statistiken**

SQL Server schätzt anhand von Statistiken, wie viele Zeilen eine Abfrage liefern wird, und wählt danach den Ausführungsplan. Die automatische Aktualisierung greift erst, wenn sich ein bestimmter Anteil der Daten geändert hat. Bei großen, langsam wachsenden Tabellen kann das dauern, und die Schätzung liegt zwischenzeitlich weit daneben. Das Ergebnis sind Pläne, die für zehn Zeilen gebaut wurden und auf zehn Millionen treffen. Eine regelmäßige, geplante Aktualisierung ist deshalb Pflicht, nicht Kür.

### **Zu wenig Arbeitsspeicher und falsche Speichergrenzen**

SQL Server lebt vom Puffercache. Was im Arbeitsspeicher liegt, muss nicht von der Platte gelesen werden. Zwei Fehler treten häufig gemeinsam auf: Die VM hat insgesamt zu wenig Arbeitsspeicher für die Datenmenge, die regelmäßig gebraucht wird, und die Einstellung „max server memory“ steht entweder auf dem Standardwert oder auf einer Zahl, die jemand vor Jahren für eine kleinere VM eingetragen hat. Im ersten Fall drängt SQL Server das Betriebssystem an die Wand, im zweiten liegt bezahlter Speicher brach.

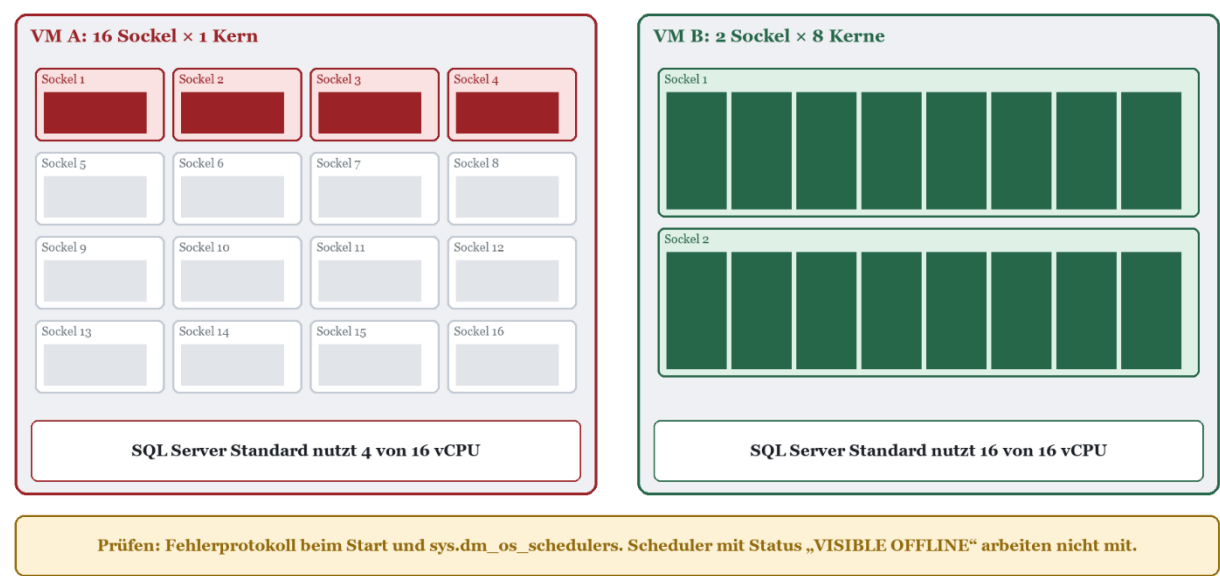
Auch hier spielt die Edition mit: Die Standard Edition begrenzt den Puffercache je Instanz. Microsoft nennt für SQL Server 2025 eine Obergrenze von 256 GB, frühere Versionen lagen niedriger. Mehr Arbeitsspeicher in der VM bringt dem Puffercache oberhalb dieser Grenze nichts mehr.

### **VM-Konfiguration: Die Edition nutzt nur einen Teil der Kerne**

Dieser Befund verdient einen Ehrenplatz, weil er so viel kostet und so leicht zu übersehen ist. Die Standard Edition ist laut Microsoft auf den kleineren Wert aus 4 Sockeln oder 24 Kernen begrenzt, ab SQL Server 2025 auf 4 Sockel oder 32 Kerne. In einer VM zählt dabei, wie die virtuellen Prozessoren präsentiert werden. Bekommt eine VM 16 vCPU als 16 Sockel mit je einem Kern, nutzt die Standard Edition davon genau vier. Die anderen zwölf stehen im Task-Manager, arbeiten aber nicht für die Datenbank. Sie sind sozusagen in Rufbereitschaft, nur ruft nie jemand an.

Skizze 2: Gleiche VM, gleiche 16 vCPU, andere Topologie

Die Standard Edition nutzt höchstens 4 Sockel. Wie die vCPU auf Sockel verteilt sind, entscheidet die VM-Konfiguration.



Vereinfachte Darstellung. Grenze laut Microsoft: kleinerer Wert aus 4 Sockeln oder 24 Kernen (bis 2022) bzw. 32 Kernen (2025).

Der Befund ist schnell bestätigt. Beim Start schreibt SQL Server ins Fehlerprotokoll, wie viele Sockel und Kerne er erkannt hat und wie viele er lizenzbedingt nutzt. In sys.dm\_os\_schedulers stehen nicht genutzte Scheduler mit dem Status „VISIBLE OFFLINE“. Die Abhilfe liegt in der VM-Konfiguration: VM herunterfahren, Topologie auf wenige Sockel mit mehreren Kernen umstellen, starten, nachmessen. Das ist eine Aufgabe der IT und braucht in der Regel keine Freigabe des Herstellers, wohl aber ein Wartungsfenster.

| Edition           | Rechenleistung je Instanz                   | Puffercache je Instanz              | SQL Server Agent |
|-------------------|---------------------------------------------|-------------------------------------|------------------|
| Enterprise        | Betriebssystem-Maximum                      | Betriebssystem-Maximum              | Ja               |
| Standard bis 2022 | kleinerer Wert aus 4 Sockeln oder 24 Kernen | begrenzt, Wert je Version prüfen    | Ja               |
| Standard 2025     | kleinerer Wert aus 4 Sockeln oder 32 Kernen | 256 GB                              | Ja               |
| Express 2025      | kleinerer Wert aus 1 Sockel oder 4 Kernen   | 1.410 MB, Datenbank höchstens 50 GB | Nein             |

Abfragen, die Terabytes lesen

Der eindrucksvollste Befund kommt meist zuletzt, weil man ihn nur sieht, wenn man die Abfragen selbst auswertet. Über den Abfragespeicher (Query Store) oder die Ausführungsstatistiken im Plancache lässt sich ermitteln, welche Abfragen wie oft laufen und wie viele Seiten sie dabei lesen. Regelmäßig finden sich einzelne Abfragen, die über den Tag gerechnet ein Vielfaches der gesamten Datenbankgröße lesen, weil sie bei jedem Aufruf große Tabellen vollständig durchsuchen. Typische Ursachen: ein fehlender Index, eine Funktion in der WHERE-Bedingung, eine Ansicht über Ansichten über Ansichten oder eine Startseite, die beim Öffnen „mal eben“ alle offenen Vorgänge der letzten zehn Jahre zählt.

Solche Abfragen sind der Grund, warum sich Anwendungen nach einem Hardwaretausch oft kaum schneller anfühlen. Wer eine Tabelle zehntausendmal am Tag vollständig liest, liest sie auf neuer Hardware eben schneller vollständig. Behoben ist damit nichts.

## Befund, Wirkung, Maßnahme und wer das darf

Die folgende Tabelle fasst die typischen Befunde zusammen. Die letzte Spalte ist die wichtigste: Sie entscheidet, ob Sie morgen handeln können oder erst den Hersteller fragen müssen.

| Befund                                              | Wirkung                                                | Maßnahme                                        | Wer darf das                               |
|-----------------------------------------------------|--------------------------------------------------------|-------------------------------------------------|--------------------------------------------|
| <b>SQL Agent gestoppt</b>                           | Keine Wartung, oft keine Protokollsicherung            | Dienst auf automatisch, Jobs und Verlauf prüfen | IT, Hersteller informieren                 |
| <b>Nie DBCC CHECKDB</b>                             | Beschädigungen bleiben unentdeckt, auch in Sicherungen | Regelmäßige Prüfung im Wartungsfenster          | IT                                         |
| <b>Veraltete Statistiken</b>                        | Falsche Pläne, schwankende Laufzeiten                  | Geplante Aktualisierung                         | IT, Vorgehen mit Hersteller abstimmen      |
| <b>Zu wenig RAM, falsches max server memory</b>     | Viel Plattenlesen, Druck auf das Betriebssystem        | RAM anpassen, Grenze setzen                     | IT                                         |
| <b>VM-Topologie passt nicht zur Edition</b>         | Bezahlte vCPU arbeiten nicht                           | Sockel und Kerne neu zuschneiden                | IT                                         |
| <b>MAXDOP und Kostenschwelle auf Standardwerten</b> | Unnötige Parallelität, Wartezeiten                     | Werte nach Microsoft-Empfehlung setzen          | IT, sofern der Hersteller nichts vorgibt   |
| <b>Alter Kompatibilitätsgrad</b>                    | Neuere Optimierungen nicht aktiv                       | Anheben nach Test                               | Nur mit Herstellerfreigabe                 |
| <b>Fehlende Indizes</b>                             | Abfragen lesen große Tabellen vollständig              | Eigener Index nach Messung                      | Nur mit Herstellerfreigabe                 |
| <b>Ineffiziente Abfrage im Code</b>                 | Dauerlast, auch auf neuer Hardware                     | Fehlermeldung an Hersteller mit Messdaten       | Hersteller                                 |
| <b>Veralteter Patchstand</b>                        | Bekannte Fehler und Lücken offen                       | Kumulative Updates einspielen                   | IT, Freigabeliste des Herstellers beachten |

## Warum Hardware allein oft nicht hilft

Die naheliegende Reaktion auf eine langsame Anwendung ist eine größere VM. Das ist verständlich: Hardware lässt sich bestellen, abrechnen und vorzeigen. Eine Statistikaktualisierung kann man der Geschäftsführung schlecht als Investition präsentieren. Trotzdem endet der Hardwareweg erstaunlich oft in einem Gespräch, das mit „Wir haben doch jetzt doppelt so viel ...“ beginnt.

## Mehr Kerne für eine Abfrage, die auf etwas anderes wartet

Eine Abfrage, die auf Sperren wartet, wartet auf neuer Hardware genauso lange. Eine Abfrage, die seriell läuft, nutzt von 32 Kernen einen. Und wenn die Standard Edition wegen der VM-Topologie nur vier Sockel sieht, verpuffen zusätzliche vCPU vollständig. Mehr Arbeitsspeicher hilft, solange er unterhalb der Editions-grenze bleibt und tatsächlich der Lesedruck das Problem ist. All das lässt sich messen, bevor man bestellt.

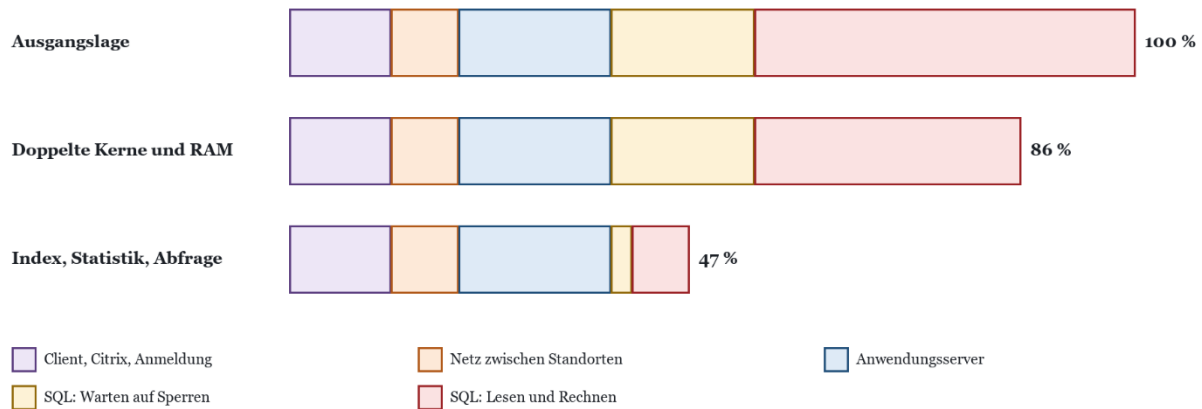
## Die Antwortzeit hat mehr Bestandteile als die Datenbank

Was am Stationsrechner als „langsam“ ankommt, ist die Summe vieler Teilstrecken: Anmeldung und Citrix-Sitzung, Netz zwischen den Standorten, Anwendungsserver, Datenbank. Viele Anwendungen schicken für eine einzige Bildschirmmaske Dutzende oder Hunderte kleiner Abfragen hintereinander. Dann dominiert nicht die einzelne Abfrage, sondern die Zahl der Hin- und Rückwege. Gerade in Klinikverbünden mit zentralem Rechenzentrum und entfernten

Standorten ist das ein häufiger Grund, warum sich eine technisch „schnelle“ Datenbank trotzdem zäh anfühlt.

### Skizze 3: Wo die Wartezeit am Arbeitsplatz entsteht

*Mehr Kerne verkürzen nur den Teil, der an Kernen hängt. Der Rest bleibt.*



Erst messen, wo die Zeit liegt. Dann dort ansetzen, nicht dort, wo es am einfachsten zu bestellen ist.

*Illustrative Anteile, keine Messwerte. Die Verteilung ist in jeder Anwendung anders.*

### TIPP · Erst die Frage, dann die Bestellung

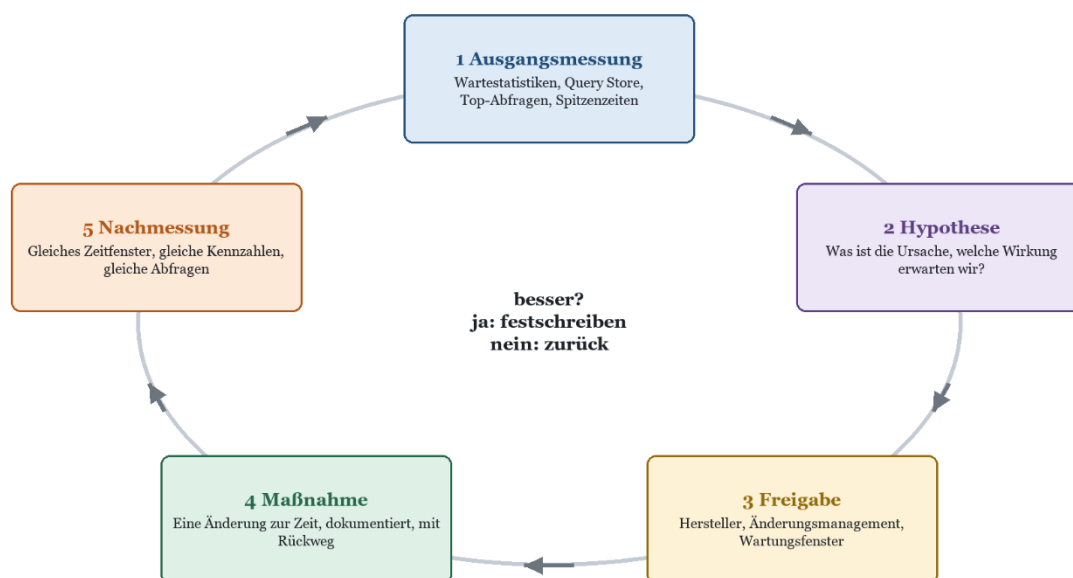
Bevor Sie Hardware beantragen, beantworten Sie drei Fragen mit Messwerten: Worauf wartet die Instanz (Wartestatistiken)? Welche Abfragen verursachen die Last (Query Store)? Wie viel der Antwortzeit am Arbeitsplatz entfällt überhaupt auf die Datenbank? Zeigen die Antworten auf echte CPU- oder Speicherengpässe, ist die Bestellung gut begründet. Wenn nicht, sparen Sie sich die Enttäuschung und das Budget.

### Messen statt fühlen: der Zyklus

„Fühlt sich schneller an“ ist im Krankenhaus eine ehrliche Rückmeldung, aber keine Messung. Nach einem Wochenende mit Wartungsarbeiten ist jede Anwendung gefühlt schneller, weil montags früh noch nicht alle angemeldet sind. Belastbar wird es erst mit einem festen Ablauf: Ausgangsmessung, Hypothese, Freigabe, eine einzelne Maßnahme, Nachmessung im gleichen Zeitfenster mit den gleichen Kennzahlen. Wer drei Dinge gleichzeitig ändert, weiß hinterher nicht, welches geholfen hat und welches er nach dem nächsten Update wiederherstellen muss.

## Skizze 4: Messung, Maßnahme, Nachmessung

Jede Änderung ist ein Experiment mit Ausgangswert, Hypothese und Vergleich.



Ohne Ausgangsmessung lässt sich hinterher nur noch über Gefühle streiten.

| Was messen                                               | Womit                                                             | Wann                                                   |
|----------------------------------------------------------|-------------------------------------------------------------------|--------------------------------------------------------|
| <b>Wartestatistiken der Instanz</b>                      | sys.dm_os_wait_stats, Differenz zweier Zeitpunkte                 | Vor und nach jeder Maßnahme, zu Spitzenzeiten          |
| <b>Teuerste Abfragen nach Lesezugriffen und Laufzeit</b> | Query Store, sys.dm_exec_query_stats                              | Über mehrere Tage, inklusive Visite und Schichtwechsel |
| <b>Genutzte Scheduler und erkannte Topologie</b>         | Fehlerprotokoll, sys.dm_os_schedulers                             | Einmalig und nach jeder VM-Änderung                    |
| <b>Speichernutzung, Verweildauer von Seiten im Cache</b> | Leistungsindikatoren                                              | Fortlaufend                                            |
| <b>Zustand der Wartung</b>                               | Jobverlauf, letzte CHECKDB, Alter der Statistiken                 | Monatlich                                              |
| <b>Antwortzeit am Arbeitsplatz</b>                       | Definierte Testschritte mit Stoppuhr oder Protokoll der Anwendung | Vor und nach, gleicher Arbeitsplatz, gleiche Uhrzeit   |

Der Abfragespeicher ist dabei das wichtigste Werkzeug. Er zeichnet Pläne und Laufzeiten je Abfrage dauerhaft in der Datenbank auf und zeigt auch, wann sich ein Plan geändert hat. Laut Microsoft ist er ab SQL Server 2022 für neue Datenbanken standardmäßig eingeschaltet. Bei älteren Datenbanken, die per Upgrade migriert sind, ist er das häufig nicht. Ob Sie ihn in der Datenbank eines Herstellers aktivieren dürfen, klären Sie vorher, denn er verändert die Datenbankkonfiguration und belegt Platz.

### AUS DER PRAXIS · Analyse einer Medizintechnik- und CAFM-Anwendung

In einem Krankenhausverbund mit mehreren Standorten stand eine Anwendung für Medizintechnik und technisches Gebäudemanagement im Ruf, „einfach langsam“ zu sein. Die Analyse folgte dem beschriebenen Zyklus und förderte genau die Muster zutage, um die es in diesem Beitrag geht.

Der SQL Agent lief nicht, Wartungsjobs waren also angelegt, aber nicht ausgeführt worden. Eine Konsistenzprüfung hatte es nicht gegeben. Viele Statistiken waren sehr alt. Die VM war so zugeschnitten, dass die eingesetzte Edition nur einen Teil der zugewiesenen Kerne nutzen konnte, und der Arbeitsspeicher war knapp bemessen.

Der eigentliche Lastverursacher waren aber wenige Abfragen, die im Lauf eines Tages Datenmengen in der Größenordnung von Terabytes lasen, bei einer Datenbank, die um Größenordnungen kleiner war. Mehr Hardware hätte das Lesen nur beschleunigt, nicht beseitigt.

Die Lehre: Die Infrastrukturbefunde konnte die IT des Hauses selbst beheben. Für Indizes und Abfragen brauchte es den Hersteller am Tisch, und dafür waren die Messdaten das beste Argument.

## Zusammenarbeit mit dem Hersteller: Freigaben, Indizes, Updates

Bei Subsystemen im Krankenhaus ist der Hersteller kein Gegner, sondern meist der Einzige, der das Datenmodell wirklich kennt. Gleichzeitig hat er ein berechtigtes Interesse daran, dass niemand an seiner Datenbank herumschraubt und ihn danach für die Folgen verantwortlich macht. Für Software in der Medizintechnik kommen oft regulatorische Anforderungen an Änderungen hinzu. Die Lösung ist deshalb kein Alleingang der IT, sondern eine geregelte Zusammenarbeit mit klaren, schriftlichen Freigaben.

### Welche Freigaben Sie schriftlich brauchen

- **Unterstützte SQL-Server-Versionen und Patchstände:** Welche Versionen und kumulativen Updates sind freigegeben, und wie schnell zieht der Hersteller nach?
- **Kompatibilitätsgrad:** Mit welchem Kompatibilitätsgrad ist die Anwendung getestet, und darf er angehoben werden?
- **Instanzeinstellungen:** Vorgaben zu MAXDOP, Kostenschwelle für Parallelität, Sortierfolge und Speichergrenzen.
- **Wartung:** Darf das Haus eigene Wartungsjobs für Indizes und Statistiken betreiben, oder liefert der Hersteller welche mit?
- **Eigene Indizes:** Unter welchen Bedingungen sind eigene Indizes erlaubt, und wie werden sie vor Updates behandelt?
- **Query Store:** Darf der Abfragespeicher in der Datenbank aktiviert werden?

### Kompatibilitätsgrad: der unterschätzte Hebel

Wird eine Datenbank auf eine neuere SQL-Server-Version migriert, behält sie ihren bisherigen Kompatibilitätsgrad. Die Instanz ist dann neu, die Datenbank verhält sich aber in vielen Punkten wie auf der alten Version. Neuere Optimierungen der Abfrageverarbeitung greifen erst mit einem höheren Kompatibilitätsgrad. Das kann deutlich helfen, es kann aber auch einzelne Abfragen verschlechtern, weil sich etwa die Schätzung der Zeilenzahlen ändert. Genau deshalb gehört die Anhebung in den Zyklus: Ausgangsmessung mit Query Store, Anhebung in einer Testumgebung oder im Wartungsfenster, Nachmessung, notfalls Rückkehr. Und davor die Freigabe des Herstellers.

### Eigene Indizes und ihr Lebenszyklus vor Herstellerupdates

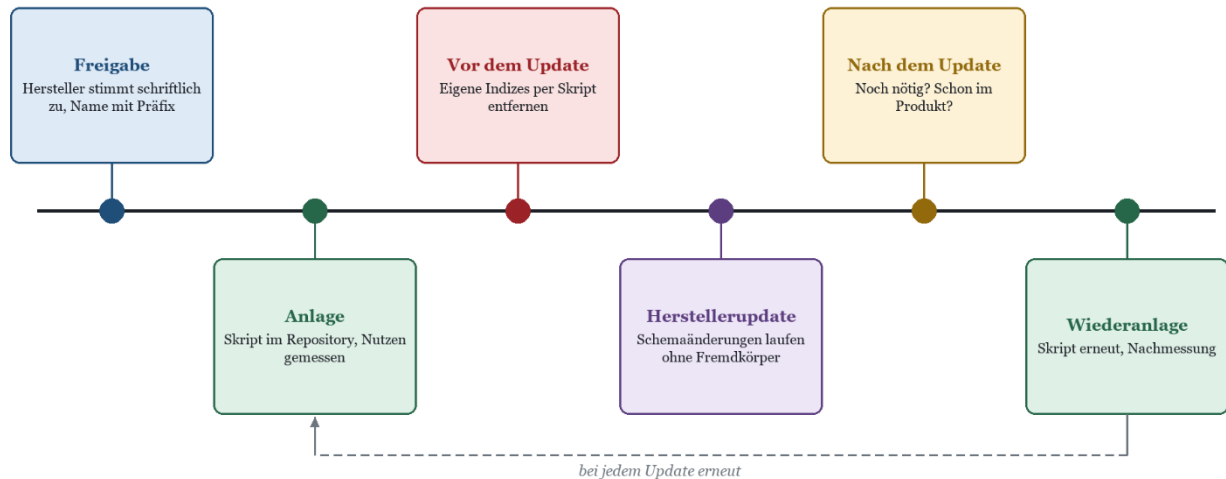
Ein gut gewählter Index kann eine Abfrage, die Gigabytes liest, auf wenige Seiten reduzieren. Er ist damit oft die wirksamste Einzelmaßnahme überhaupt. Er ist aber auch ein Fremdkörper im Datenmodell des Herstellers. Herstellerupdates ändern Tabellen, benennen Spalten um oder legen selbst Indizes an. Ein eigener Index kann ein solches Update im schlechtesten Fall abbrechen lassen, und dann steht das Subsystem mitten im Wartungsfenster still.

Bewährt hat sich ein fester Lebenszyklus: Eigene Indizes bekommen ein erkennbares Namenspräfix, werden als Skript in einem Repository gepflegt und mit Begründung und Messwert

dokumentiert. Vor jedem Herstellerupdate werden sie per Skript entfernt, nach dem Update geprüft und nur dann wieder angelegt, wenn sie noch gebraucht werden. Manchmal hat der Hersteller das Problem inzwischen selbst gelöst. Das ist der schönste Moment im Leben eines eigenen Indexes: Er darf in Rente.

### Skizze 5: Lebenszyklus eines eigenen Indexes rund um ein Herstellerupdate

*Eigene Indizes sind tragbar, solange sie erkennbar, dokumentiert und rückbaubar sind.*



*Schema für eigene Indizes. Den konkreten Ablauf legen Hersteller und Haus gemeinsam fest.*

### WICHTIG · Supportfähigkeit nicht verspielen

Änderungen ohne Freigabe können den Supportanspruch gegenüber dem Hersteller gefährden. Für ein Subsystem, das in der Patientenversorgung gebraucht wird, ist ein „Diese Konfiguration unterstützen wir nicht“ am Freitagabend das Letzte, was Sie brauchen. Holen Sie jede Freigabe schriftlich ein und legen Sie sie zur Systemdokumentation.

### Versionsstand und Pflichten zur IT-Sicherheit

Langsame Subsysteme laufen auffallend oft auf alten Versionen. Für SQL Server 2016 ist laut Microsoft-Lebenszyklusrichtlinie im Juli 2026 der erweiterte Support ausgelaufen. Sicherheitsupdates gibt es danach nur noch über kostenpflichtige erweiterte Sicherheitsupdates. Die Versionsfrage ist damit nicht nur eine Performance-, sondern auch eine Sicherheitsfrage. Wer die Instanz im Zuge der Analyse ohnehin anfasst, sollte den Weg zu einer unterstützten Version gleich mit dem Hersteller klären.

Auch die Anmeldung an der Instanz gehört auf den Prüfstand. Viele Subsysteme verbinden sich noch per NTLM oder mit SQL-Konten, deren Kennwort seit dem Produktivstart unverändert ist. Wie Klinikapplikationen sauber auf Kerberos umziehen, beschreibt der Beitrag [NTLM abschalten im Krankenhaus: Medizingeräte, Altanwendungen und Kerberos](#). Die technischen Grundlagen finden Sie im [Kompetenzbereich Kerberos](#).

### HINWEIS · Kein Ersatz für eine Rechtsberatung

Die IT-Sicherheit im Krankenhaus regelt § 391 SGB V, seit dem NIS2-Umsetzungsgesetz vom 6. Dezember 2025 in neuer Fassung. Für Betreiber kritischer Anlagen gelten stattdessen die Pflichten des BSIG. Der branchenspezifische Sicherheitsstandard „Medizinische Versorgung“ der DKG liegt in Version 1.3.1 vom 10. Oktober 2025 vor. Welche konkreten Anforderungen sich daraus für Wartung,

Patchstand und Verfügbarkeit Ihrer Subsysteme ergeben, hängt von der Einstufung Ihres Hauses ab und gehört mit dem Informationssicherheitsbeauftragten und gegebenenfalls juristischer Beratung geklärt. Die Einordnung für Microsoft 365 finden Sie im Beitrag [NIS2, § 391 SGB V und B3S: Was die Sicherheitspflichten für Microsoft 365 bedeuten](#).

### **Betrieb im Alltag: Rufbereitschaft, Downtime, Notbetrieb**

Am Ende entscheidet sich die Performance eines Subsystems nicht in der Analyse, sondern im Betrieb. Dazu gehören eine Überwachung, die vor der Station merkt, dass der Agent steht, ein Ablauf für die Rufbereitschaft, der auch die Datenbank umfasst, und ein Wartungsfenster, das mit den Funktionsbereichen abgestimmt ist. Wie eine Klinik den Ausfall eines Systems übt, beschreibt der Beitrag [Notfallübung für die Klinik-IT: Downtime, Notbetrieb und die Rückkehr zum Papier](#). Bei einer CAFM-Anwendung in der Medizintechnik hängt zudem die Nachweisführung an deren Verfügbarkeit, mehr dazu unter [Medizinproduktebuch, Einweisungen, Gerätenachweise: MPBetriebV-Dokumentation zwischen CAFM und SharePoint](#).

## **Häufige Fragen zur SQL Server Performance im Krankenhaus**

### **Darf die IT den SQL Server eines Subsystems überhaupt anfassen?**

Für die Infrastruktur in der Regel ja: VM-Topologie, Arbeitsspeicher, Dienste, Konsistenzprüfung und Sicherung liegen meist im Verantwortungsbereich des Hauses. Alles, was das Datenmodell oder das Verhalten der Anwendung verändert, also eigene Indizes, Kompatibilitätsgrad oder Instanzeinstellungen mit Vorgaben des Herstellers, braucht eine schriftliche Freigabe. Klären Sie die Abgrenzung einmal sauber und halten Sie sie in der Systemdokumentation fest.

### **Woran erkenne ich, ob die VM-Konfiguration Kerne verschenkt?**

Im Fehlerprotokoll der Instanz steht beim Start, wie viele Sockel und Kerne SQL Server erkannt hat und wie viele er nutzt. In sys.dm\_os\_schedulers erscheinen nicht genutzte Scheduler mit dem Status „VISIBLE OFFLINE“. Betroffen ist vor allem die Standard Edition, wenn die VM viele Sockel mit je einem Kern präsentiert.

### **Hilft es, einfach auf die Enterprise Edition zu wechseln?**

Manchmal, aber selten als erste Maßnahme. Die Enterprise Edition hebt die Grenzen bei Kernen und Arbeitsspeicher auf und bringt zusätzliche Funktionen der Abfrageverarbeitung mit. Abfragen, die wegen eines fehlenden Indexes große Tabellen vollständig lesen, bleiben aber teuer. Erst messen, dann über die Edition entscheiden, und vorab klären, welche Edition der Hersteller unterstützt.

### **Wie oft sollte eine Konsistenzprüfung laufen?**

So oft, dass eine Beschädigung entdeckt wird, solange noch eine saubere Sicherung existiert. In der Praxis heißt das mindestens wöchentlich, bei kritischen Subsystemen häufiger, abgestimmt auf die Aufbewahrungsdauer der Sicherungen und das Wartungsfenster.

### **Was mache ich, wenn der Hersteller eigene Indizes grundsätzlich ablehnt?**

Dann liefern Sie ihm die Messdaten: die Abfrage, ihre Häufigkeit, die gelesenen Seiten und den Vorschlag für einen Index samt erwarteter Wirkung. Viele Hersteller übernehmen gut begründete Indizes in ein späteres Update. Das dauert länger als ein eigener Index, ist aber dauerhaft und supportfähig.

### **Was hat das alles mit Microsoft 365 zu tun?**

Direkt wenig, indirekt viel. Die Serie zieht eine klare Grenze: Patientendaten gehören ins KIS und seine Subsysteme, Microsoft 365 dient Verwaltung, Organisation, Qualitätsmanagement und

Zusammenarbeit. Je zuverlässiger die Subsysteme laufen, desto geringer ist die Versuchung, Daten in Excel-Listen auszuleiten und in Teams abzulegen. Dass es solche Listen gibt, weiß jeder, der einmal einen Teams-Kanal einer Station gesehen hat. Normal sollten sie nicht werden. Mehr dazu im Beitrag [Patientendaten und Microsoft 365: Wo das KIS endet und das Office anfängt](#).

## **Fazit: Die Datenbank braucht einen Zuständigen, keine Wunder**

Langsame Subsysteme sind im Krankenhaus selten ein Hardwareproblem und fast nie ein Rätsel. Sie sind das vorhersehbare Ergebnis einer Datenbank, die zwischen Hersteller und IT niemandem gehört. Die Befunde ähneln sich von Haus zu Haus: gestoppter Agent, fehlende Konsistenzprüfung, alte Statistiken, knapper Speicher, eine VM, die ihre Kerne vor der Edition versteckt, und eine Handvoll Abfragen, die jeden Tag ein Vielfaches dessen lesen, was die Datenbank überhaupt enthält.

Was wirklich hilft, ist unspektakulär: eine klare Zuständigkeit, ein Zyklus aus Messung, Maßnahme und Nachmessung, schriftliche Freigaben des Herstellers und ein geregelter Umgang mit eigenen Indizes. Danach darf auch wieder über Hardware geredet werden, dann aber mit Zahlen statt Bauchgefühl. Die Station merkt den Unterschied, und die IT bekommt einen Anruf weniger. Im Krankenhaus ist das fast schon ein Grund zum Feiern.

Wenn Sie eine solche Analyse für eine Ihrer Fachanwendungen angehen möchten, finden Sie Vorgehen und Ansprechpartner unter [Microsoft 365 Beratung für Krankenhäuser](#). Die technischen Grundlagen bündelt der [Kompetenzbereich SQL Server](#).

### **WEITERLESEN · Weiterlesen**

- › [Kompetenzbereich SQL Server](#)
- › [Microsoft 365 im Krankenhaus](#)
- › [Microsoft 365 im Krankenhaus: Warum die Klinik anders tickt als Verwaltung und Mittelstand](#)
- › [NTLM abschalten im Krankenhaus: Medizingeräte, Altanwendungen und Kerberos](#)
- › [Notfallübung für die Klinik-IT: Downtime, Notbetrieb und die Rückkehr zum Papier](#)
- › [Microsoft 365 Schulung für Krankenhäuser](#)