

Monitoring

Wenn es mal im SQL-Server hakt und die aktiven Aktivitäten und Abfragen geprüft werden müssen, können folgende Befehle helfen.

Zum einen das Skript `sp_whoisactive` und zum anderen `sp_who2`. Während `sp_who2` bereits vorhanden ist, muss `sp_whoisactive` nachinstalliert werden, da dieses Skript nicht von Microsoft stammt. `sp_whoisactive` zeigt alle gerade vom SQL Server ausgeführten Abfragen an, mit Details wie z. B. durch welche andere Abfrage sie blockiert werden oder wie lange sie schon laufen. `sp_who2` zeigt diverse Informationen über alle aktiven Verbindungen zum SQL Server an. Beide Skripte werden im folgenden vorgestellt.

sp_whoisactive

Um die aktiven Prozesse auf einem Microsoft SQL-Server anzeigen zu können, bietet sich das Skript `sp_whoisactive` an. Jedoch handelt es sich nicht um einen Standard-Befehl, sondern dieser Befehl muss erst mittels eines Skriptes nachgerüstet werden.

Dazu einfach den Source Code von der [Github Seite](#) herunterladen und entpacken. Dann z. B. das *Microsoft SQL Server Management Studio* öffnen und die Master Datenbank als aktive Datenbank markieren. Danach das Skript `sp_WhoIsActive.sql` in der Master Datenbank ausführen.

Der Befehl muss in der Datenbank Master ausgeführt werden, ansonsten steht dieser nur in der Datenbank zur Verfügung, in der das Skript ausgeführt wurde, da nur dort `sp_whoisactive` als Stored Procedure gespeichert wird.

Nun steht der Befehl in allen Datenbanken zur Verfügung und kann einfach in einer beliebigen SQL-Query ausgeführt werden mit `sp_whoisactive`. Eine vollständige Dokumentation des Befehls ist hier zu finden: [sp_whoisactive Doku](#)

sp_who2

Um die aktiven Verbindungen eines SQL-Servers einsehen zu können, kann der Befehl `sp_who2` verwendet werden. Es können mehrere Verbindungen pro Rechner und User sein. Deswegen empfiehlt es sich folgende Abfrage zu verwenden, um die Informationen etwas aufzubereiten.

```
CREATE TABLE #sp_who2 (  
    [SPID] INT, Status VARCHAR(255)  
    [], Login VARCHAR(255)
```

```

[,HostName VARCHAR(255)
[,BlkBy VARCHAR(255)
[,DBName VARCHAR(255)
[,Command VARCHAR(255)
[,CPUTime INT
[,DiskIO INT
[,LastBatch VARCHAR(255)
[,ProgramName VARCHAR(255)
[,SPID2 INT
[,REQUESTID INT
)
INSERT INTO #sp_who2 EXEC sp_who2
SELECT
[,Login
[,HostName
[,DBName
[,Command
[,ProgramName
[,COUNT(*) AS Verbindungen
[,SUM(CPUTime) AS CPUTime
[,SUM(DiskIO) AS DiskIO
[,MAX(LastBatch) AS LastBatch
FROM #sp_who2
WHERE SPID > 50 AND DBName = 'Testdatenbank'
GROUP BY
[,Login
[,HostName
[,DBName
[,Command
[,ProgramName
ORDER BY Login ASC
DROP TABLE #sp_who2

```

Der Befehl `sp_who2` wurde um eine temporäre Tabelle erweitert, in welche die ermittelten Daten geschrieben werden. Dadurch steigt zwar der Aufwand für die Ausführung, jedoch können die Daten so sinnvoller gefiltert werden, da sonst sehr viele Daten dargestellt werden. In diesem Beispiel werden einige Systemprozesse ausgeschlossen und nur Verbindungen zur Datenbank *Testdatenbank* angezeigt. Zusätzlich wird gruppiert, um mehrfache Verbindungen zusammenzufassen.