Ola Hallengren Datenbankwartung einrichten

Description

In diesem Artikel beschreibe ich, wie eine automatisierte Datenbankwartung fÃ¼r einen Microsoft SQL Server eingerichtet werden kann.

HierfÃ¼r nutzen wir die in der Praxis weit verbreiteten Wartungsskripte von Ola Hallengren. Diese stellen eine Sammlung von Stored Procedures bereit, mit denen sich Wartungsaufgaben wie Backups, Indexoptimierung und IntegritÃ¤tsprÃ¼fungen automatisiert durchfÃ¼hren lassen.

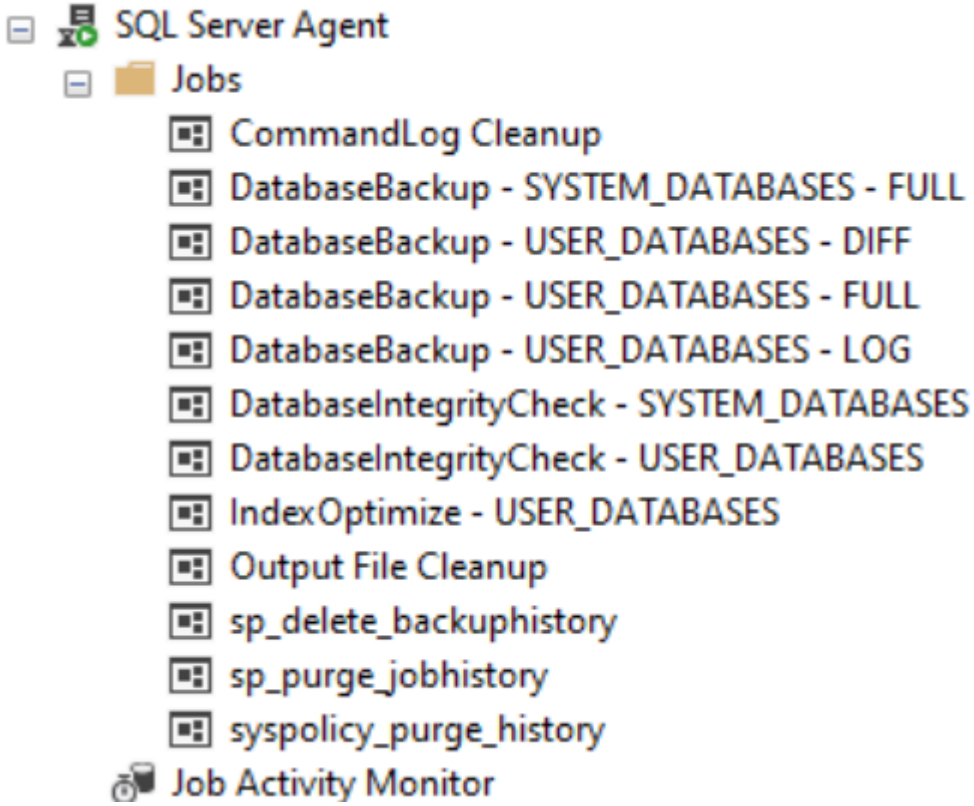
Auf Basis dieser Procedures kÃ¶nnen SQL Server Agent Jobs erstellt werden, die die gewÃ¼nschten Wartungsaufgaben in festgelegten Intervallen ausfÃ¼hren. Die Konfiguration erfolgt Ã¼ber verschiedene Parameter, die individuell an die jeweilige Umgebung angepasst werden kÃ¶nnen.

Im ersten Schritt mÃ¼ssen wir uns diese Prozeduren erstmal installieren. Dazu gehen wir auf die folgende Website und laden uns das Skript herunter: https://ola.hallengren.com/

Sobald wir das Skript heruntergeladen haben, Ã¶ffnen wir dieses im SSMS (SQL Server Management Studio). Dort passen wir jetzt folgende Dinge an:

@BackupDirectory	Pfad in dem die Backups abgelegt werden sollen
@CleanupTime	Anzahl der Stunden wie lange die Backups auf der Platte liegen sollen. (Die Backups sollten vorher mit einer Backupsoftware an eine andere Stelle kopiert werden).
USE [master]	Wir erstellen eine neue Datenbank und Ã¤ndern **master** auf unsere neue **Datenbank**

Sobald wir diese Dinge angepasst haben, kÃ¶nnen wir das Skript ausfÃ¼hren. Wir sollten im Anschluss dann unter **Jobs** im SSMS unsre neuen Wartungsjobs sehen.



Wenn die Auflistung aussieht wie auf dem Screenshot zusehen, dann haben wir bis hierher alles richtig gemacht! In der unten stehenden Tabelle, erklÃ¤re ich kurz die Funktion der Prozedur / Job.

Job-Name	**Beschreibung**
CommandLog Cleanup	Dieser Job lÃ¶scht alle EintrÃ¤ge in der CommandLog Tabelle die Ã¤lter als 30 Tage sind.
DatabaseBackup â?? SYSTEM_DATABASE â?? FULL	Dieser Job erstellt ein Voll-Backup von allen Systemdatenbanken. Dies sind die *master*, *model* und die *msdb* Datenbank.
DatabaseBackup â?? USER_DATABASES â?? DIFF	Dieser Job erstellt von allen Benutzer-Datenbanken ein differenzielles Backup.
DatabaseBackup â?? USER_DATABASES â?? FULL	Dieser Job erstellt von allen Benutzerdatenbanken ein Voll-Backup.
DatabaseBackup â?? USER_DATABASES â?? LOG	Dieser Job erstellt ein Backup der Transaktionsprotokolle der Benutzerdatenbanken wenn diese als Wiederherstellungsmodell â??VollstÃ¤ndigâ?? ausgewÃ¤hlt haben.
DatabaseIntegrityCheck â?? SYSTEM_DATABASES	Dieser Job fÃ¼hrt ein IntegritÃ¤tscheck der Systemdatenbanken durch
DatabaseIntegrityCheck â?? USER_DATABASES	Dieser Job fÃ¼hrt ein IntegritÃ¤tscheck der Benutzerdatenbanken durch
IndexOptimize â?? USER_DATABASES	Dieser Job fÃ¼hrt eine Indexoptimierung fÃ¼r alle Benutzerdatenbanken durch

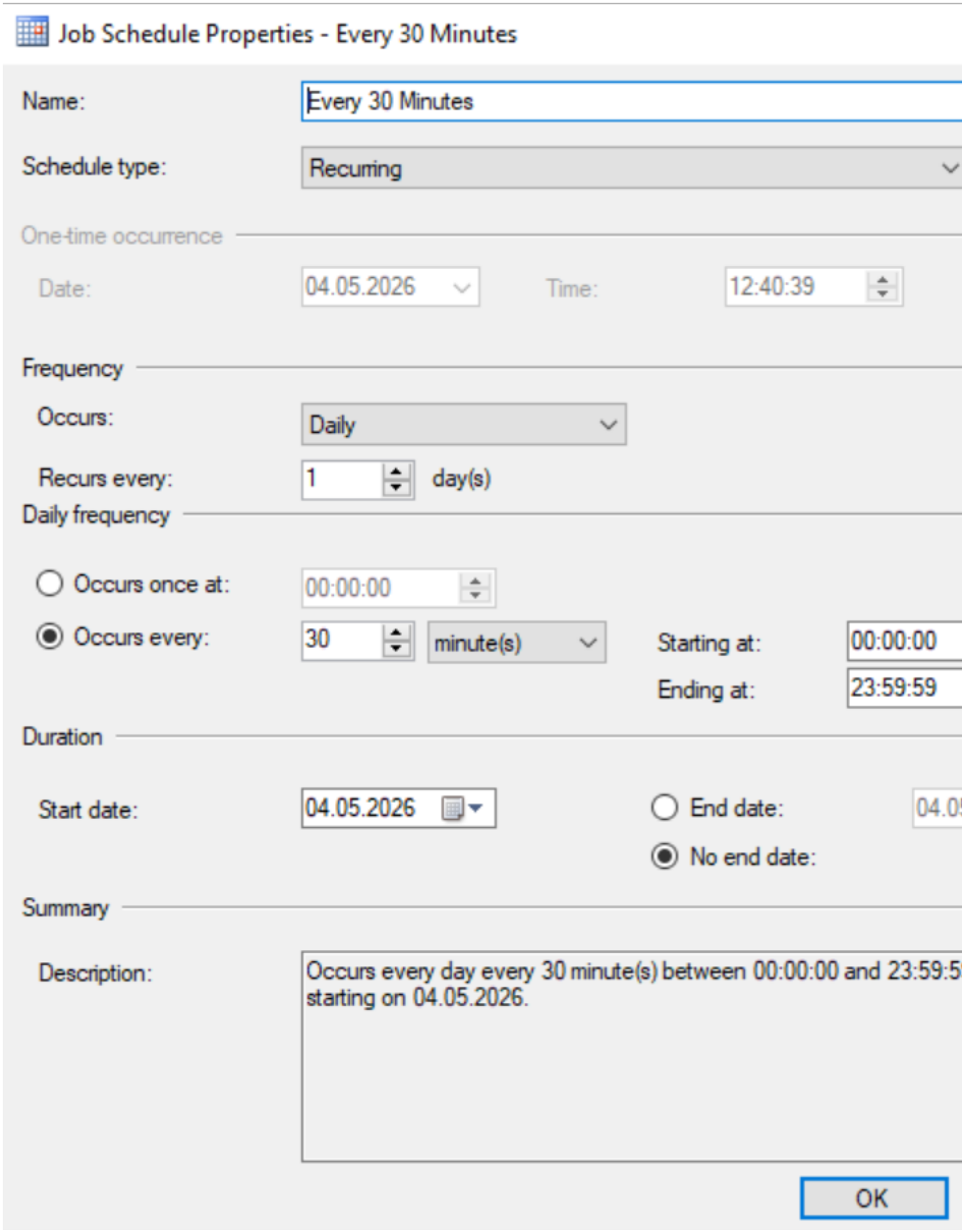
Output File Cleanup	Dieser Job lÃ¶scht alle *.txt im LOG-Verzeichnis des SQL-Servers, welche Ã¤lter als 30 Tage sind.
sp_delete_backuphistory	Dieser Job lÃ¶scht alle EintrÃ¤ge der Backup Logs in der msdb Datenbank welche Ã¤lter als 30 Tage sind.
sp_purge_jobhistory	Dieser Job lÃ¶scht den Verlauf des SQL Server Agent in der msdb Datenbank welche Ã¤lter als 30 Tage sind.

Diese Jobs mÃ¼ssen jetzt nur noch terminiert werden. FÃ¼r meinen Datenbankserver laufen die Jobs nachfolgendem Intervall:

Job-Name	**Vorgeschlagener Intervall**
DatabaseBackup â?? SYSTEM_DATABASE FULL	Einmal in der Woche
DatabaseBackup â?? USER_DATABASE DIFF	Alle 6 Stunden
DatabaseBackup â?? USER_DATABASE FULL	Einmal am Tag
DatabaseBackup â?? USER_DATABASE LOG	Alle 30 Minuten
DatabaseIntegrityCheck â?? SYSTEM_DATABASE	Einmal in der Woche
DatabaseIntegrityCheck â?? USER_DATABASES	Einmal in der Woche
IndexOptimize â?? USER_DATABASES	Einmal in der Woche
Output File Cleanup	Einmal in der Woche
sp_delete_backuphistory	Einmal in der Woche
sp_purge_jobhistory	Einmal in der Woche

Info: Es handeln sich hier um Werte die bei mir ganz gut klappen. Diese Werte mÃ¼ssen je nach Bedarf natÃ¼rlich angepasst werden.

Um den Intervall festzulegen, Ã¶ffnen wir Ã¼ber die **Eigenschaften** des Jobs den â??Schedulerâ??. Dort legen wir einen Eintrag an und definieren unseren Intervall. Sobald wir auf OK drÃ¼cken, ist der Intervall gesetzt und der Job wÃ¼rde beim Eintreffen des Intervalls anfangen die Wartungen durchzufÃ¼hren.


Job Schedule Properties - Every 30 Minutes
Name:
Every 30 Minutes
Schedule type:
Recurring
One-time occurrence
Date:
04.05.2026
Time:
12:40:39
Frequency
Occurs:
Daily
Recurs every:
1
day(s)
Daily frequency
Occurs once at:
00:00:00
Occurs every:
30
minute(s)
Starting at:
00:00:00
Ending at:
23:59:59
Duration
Start date:
04.05.2026
End date:
No end date:
Summary
Description:
Occurs every day every 30 minute(s) between 00:00:00 and 23:59:5
starting on 04.05.2026.
OK

Ã?berwachung

Um den Status der Jobs zu Ã¼berwachen, kann der folgende SQL-Query verwendet werden:

```
SELECT ID
 ,DatabaseName
 ,CommandType
 ,StartTime
 ,EndTime
 ,ErrorNumber
FROM Database.dbo.CommandLog
WHERE ErrorNumber > 0
```

Category

1. Datenbanken
2. Microsoft SQL

Date Created
04.05.2026
Author
administrator