



tse:nit banking | Hinweise zur
Datensicherung und zum
Einsatz von tse:nit banking mit
Microsoft® SQL Server™

—
gültig ab tse:nit banking
Version 2.1.4



tse:nit

Mehr Produktivität. Mehr Erfolg.



Wolters Kluwer | Software+Services
Deutschland

Maximizing Value for Customers

Inhalt

1	Einleitung.....	3
2	Hinweise zur Datensicherung	4
2.1	Umfang der zu sichernden Daten	4
2.2	Synchronität der Sicherung	4
2.3	Allgemeines	4
2.4	Häufigkeit der Sicherung	5
2.5	Methoden der Sicherung	5
2.5.1	Bewegungsdaten	5
2.5.2	SQL Server-Datenbank	5
2.6	Empfohlene Vorgehensweisen zur Sicherung von SQL Server-Datenbanken	8
2.6.1	Manuelle Sicherung mit dem SQL Server Enterprise Manager	8
2.6.2	Automatisches Sicherungsverfahren	9
2.6.3	Datenbank Wartungsplan	11
3	Vorgehensweise zur Rücksicherung der Datenbank	17
3.1	Rücksicherung der Datenbanksicherungsdatei	17
3.2	Rücksicherung der Datenbankdatei und des Transaktionsprotokolls	20
4	Hinweise zum Einsatz von tse:nit mit Microsoft® SQL Server™	21
4.1	Voraussetzungen	21
4.1.1	Service-Pack-Versionen	21
4.1.2	Administrator-Passwort	22
4.1.3	Enterprise Manager	22
4.2	Eigenschaften der Datenbank	23
4.3	Wiederherstellungsmodell	24
4.4	Verkleinern der Datenbankdateien beim SQL Server	25
4.4.1	Datenbank verkleinern	25
4.4.2	Weiteres Vorgehen	26
5	Performancesteigerung	27
5.1	Voraussetzungen	27
5.1.1	SQL Server	27
5.1.2	Betriebssystem	27
5.1.3	Netzwerkconfiguration	28
5.2	Optimierung	28
5.2.1	SQL Server - Allgemein	28
5.2.2	Optimierung der SQL Server - Einstellungen	28
5.2.3	SQL Server - Eigenschaften	29
5.2.4	Netzwerkoptimierung des Servers	30
5.2.5	Netzwerkoptimierung Client	31

1 Einleitung

Dieses Dokument gibt Hinweise zur Datensicherung und zum Einsatz von Microsoft® SQL Server™ 2000 (SQL Server 2000), wenn dieser in Verbindung mit tse:nit eingesetzt wird. Es werden außerdem Hinweise für die Datensicherung der Microsoft SQL Server Desktop Engine (MSDE 2000) gegeben.

Dieses Dokument ersetzt nicht die von Microsoft® oder Drittherstellern gelieferten Dokumentationen zu angesprochenen Themen.

Sollten die Empfehlungen dieses Dokumentes von denen der Dokumentation ⇒ SYSTEMVORAUSSETZUNGEN 10IT abweichen, so gelten die Angaben der zuletzt genannten Dokumentation als Mindestvoraussetzung (insbesondere hinsichtlich der notwendigen Service Packs).

Bitte beachten Sie, dass zum ordnungsgemäßen Betrieb von tse:nit bestimmte Einstellungen zwingend notwendig sind. Andere Einstellungen und Maßnahmen sind Vorschläge, die Sie auf Ihre individuellen Bedürfnisse anpassen können oder sogar müssen.

Bitte betrauen Sie einen Systembetreuer mit der Herstellung der genannten Einstellungen und Maßnahmen, um kostenintensive Fehler zu vermeiden.

Da es sich um sehr umfangreiche Themen handelt, möchten wir Ihnen mit dieser Dokumentation einen ersten Überblick geben.

2 Hinweise zur Datensicherung

Die ADDISON Software und Service GmbH übernimmt keine Haftung für Schäden, die durch den Verlust von Daten entstehen. Die Verantwortung für die regelmäßige Datensicherung liegt beim Kunden. Sowohl beim Einzelplatz als auch im Netzwerk muss eine externe Datensicherung vorhanden sein. Es müssen mindestens die tse:nit *banking* Datenbank und die damit korrespondierenden Bewegungsdaten gesichert werden.

In diesem Kapitel werden die Besonderheiten und verschiedene Methoden zur Datensicherung vorgestellt.

2.1 Umfang der zu sichernden Daten

Zur Sicherung des gesamten tse:nit Datenbestandes müssen folgende Daten gesichert werden:

- **Datenbank**, die von einem SQL Server 2000 verwaltet wird.

Die Standardbezeichnung ist **db_zp_10it_banking**.

Die Datenbank **db_zp_10it_banking** besteht aus der Datenbankdatei **db_zp_10it_banking.mdf** und dem Transaktionsprotokoll **db_zp_10it_banking_log.ldf**. Beide Dateien müssen gesichert werden.

Außerdem empfehlen wir, zusätzlich zur tse:nit *banking*-Datenbank auch die Datenbanken **master** und **msdb** zu sichern, damit bei einer Neuinstallation des SQL Servers die tse:nit *banking* Datenbank ohne weitere auszuführende Befehle und Prozeduren zurückgesichert werden kann.

- Verzeichnisse der **Bewegungsdaten** einschließlich der darin befindlichen Unterverzeichnisse.

Die Standardbezeichnung ist **10it_Banking_Daten (Freigabename)**.

Zu den Bewegungsdaten zählen die Dateien in folgenden Unterverzeichnissen:

- ⇒ Backup - Kopien der DB_zp_10it_Banking (erstellt mit dem tse:nit *banking* administrations tool)
- ⇒ TEMP - internes Verzeichnis für die Ablage temporärer Dateien
- ⇒ XMLExport - Exportpfad der Kontoumsätze für den tse:nit Bankauszug



Bei einer Einzelplatzinstallation können die o. a. Verzeichnisse im Programmverzeichnis von tse:nit *banking* vorhanden sein.

2.2 Synchronität der Sicherung

Um eine Synchronität zwischen den in der Datenbank enthaltenen Daten und den Bewegungsdaten zu erreichen, sollte die Sicherung zu einem Zeitpunkt erfolgen, an dem keine Änderungen in den Bewegungsdaten vorgenommen werden. Im Idealfall bedeutet das, dass kein Mitarbeiter mehr in tse:nit *banking* arbeitet, obwohl es bei Verwendung der von uns empfohlenen Sicherungsvariante nicht grundsätzlich erforderlich ist.

2.3 Allgemeines

- Sicherungen sollten außer Hause und sicher (z.B. Bankschließfach) aufbewahrt werden.
- Sicherungen auf gleicher Festplatte bzw. Partition sind nutzlos, wenn die Festplatte zerstört ist.
- Bewahren Sie ältere Sicherungen für eine bestimmte Zeit auf, falls die neueste Sicherung beschädigt oder zerstört wird oder verloren geht oder Probleme erst später bemerkt werden.
- Erstellen Sie ein System zum Überschreiben von Sicherungen, bei dem die ältesten Sicherungen zuerst wieder verwendet werden.

- Verwenden Sie Ablaufdaten für Sicherungen, um verfrühtes Überschreiben zu vermeiden.
- Beschriften Sie die Sicherungsmedien, damit wichtige Sicherungen nicht versehentlich überschrieben werden. Dies ermöglicht eine problemlose Identifizierung der auf den Sicherungsmedien gespeicherten Daten oder eines bestimmten Sicherungssatzes.
- Testen Sie Ihre Sicherungsroutine. Häufig wird erst im Ernstfall festgestellt, dass die Sicherungsroutine nicht oder sogar nie funktioniert hat oder der gesicherte Datenbestand selbst schon defekt war.
- Reden Sie über die für Sie geeignete Sicherungsroutine mit Ihrem Systembetreuer.

2.4 Häufigkeit der Sicherung

Wir empfehlen, den Datenbestand mindestens einmal täglich komplett zu sichern. Darüber hinaus muss unmittelbar vor dem Einspielen einer neuen Version von tse:nit bzw. tse:nit *banking* eine Sicherung des tse:nit Datenbestandes und des tse:nit *banking*-Datenbestandes durchgeführt werden.

Bitte beachten Sie, dass für Sicherungsmedien in Abhängigkeit z.B. vom Umfang des Datenbestandes und der täglich anfallenden Änderungen am Datenbestand Aufbewahrungsfristen vorgesehen werden sollten.

So ist zum Beispiel folgende Staffelung denkbar:

- tägliche Sicherung über 2 Wochen zurücklegen,
- wöchentliche Sicherung über einen Monat zurücklegen,
- Monatssicherung über ein Jahr zurücklegen,
- eine Jahressicherung aufbewahren.

Des Weiteren kann es sinnvoll sein, Transaktionsprotokollsicherungen in stündlichen Abständen durchzuführen, wenn Daten in Massen geändert bzw. angelegt werden.

Die Häufigkeit der Datensicherung bei einer **Einplatzinstallation** hängt von der Veränderung Ihrer Daten ab. Da diese vermutlich nicht so hoch wie bei einer Mehrplatzinstallation ist, wird eine tägliche Sicherung nicht zwingend notwendig sein. Berücksichtigen Sie jedoch den notwendigen Aufwand zur Wiederherstellung der verlorenen Daten, wenn Ihre Sicherung zum Beispiel mehrere Tage alt ist.

Aber auch hier gilt: Bitte erstellen Sie Sicherungen generationsweise und bewahren Sie diese über einen längeren Zeitraum auf.

2.5 Methoden der Sicherung

2.5.1 Bewegungsdaten

Die Bewegungsdaten können zum Beispiel mit einer gängigen Sicherungs-Software gesichert werden. Es wird empfohlen, das Verzeichnis der Bewegungsdaten komplett inkl. aller Unterverzeichnisse zu sichern.

2.5.2 SQL Server-Datenbank

Beim Microsoft SQL Server 2000 besteht eine Datenbank aus zwei Dateien, die sich im Unterverzeichnis DATA des während der Installation des Microsoft SQL Server angegebenen Datenverzeichnisses befinden. Diese Dateien heißen DB_ZP_10IT_BANKING.MDF (Datenbankdatei) und DB_ZP_10IT_BANKING_LOG.LDF (Transaktionsprotokoll). Die Dateien befinden sich bei einer Standardinstallation im Datenverzeichnis C:\PROGRAMME\MICROSOFT SQL SERVER\MSSQL\DATA (Neuinstallation von SQL Server 2000), C:\MSSQL7\DATA (Update von SQL Server 7.0 auf SQL SERVER 2000) oder C:\PROGRAMME\MICROSOFT SQL SERVER\MSSQL\$ADDISON_AKTE\DATA (Installation der MSDE 2000 über tse:nit *banking* administration tools).

Diese Dateien können nicht direkt mit einer externen Sicherungssoftware gesichert werden, solange der Dienst MSSQLServer läuft.

Zum Durchführen der Sicherung der Datenbank sind grundsätzlich drei Varianten möglich:

1. Interne Sicherung der Datenbank inklusive des Transaktionsprotokolls über den Enterprise Manager auf Festplatte und Sichern der bei dieser Sicherung erstellten Dateien mit einer externen Sicherungssoftware. Dieses ist die von uns empfohlene Methode und in Kapitel 2.6 werden entsprechende Verfahren näher beschrieben.
2. Wenn Sie gängige Sicherungssoftware zur Sicherung der Datenbank nutzen wollen, bieten Ihnen diese Programme eine Erweiterung um einen sog. SQL SERVER-AGENTEN, die auch ohne weitere Maßnahmen des Anwenders das Sichern von SQL Server-Datenbanken ermöglichen. Das in Methode 3 angesprochene Beenden und Starten der Dienste wird hier automatisch ausgeführt.

Wenn Sie eine derartige Sicherungsmethode nutzen wollen, lassen Sie sich durch einen Fachmann über Einsatzmöglichkeiten und Voraussetzungen informieren.

3. Beenden der Dienste MSSQLServer (im SQL Server-Dienst-Manager als SQL Server angezeigt) und SQLServerAgent vor dem Start der Sicherung und erneutes Starten nach dem Beenden der Sicherung, um die Datenbankdatei DB_ZP_10IT_BNKING.MDF und die LOG Datei DB_ZP_10IT_BANKING_LOG.LDF direkt zu sichern.

Geben Sie in START | AUSFÜHREN... folgende Befehle ein.

SQL Server 2000

vor der Sicherung:

```
NET STOP SQLSERVERAGENT  
NET STOP MSSQLSERVER
```

nach der Sicherung:

```
NET START MSSQLSERVER  
NET START SQLSERVERAGENT
```

Wenn Sie als Datenbank **MSDE 2000** nutzen und bei der Installation den Standardvorschlag bei NAME DER INSTANZ auf ADDISON_AKTE belassen haben, dann heißt der Befehl:

vor der Sicherung:

```
NET STOP SQLAgent$ADDISON_AKTE  
NET STOP MSSQL$ADDISON_AKTE
```

und nach der Sicherung:

```
NET START MSSQL$ADDISON_AKTE  
NET START SQLAgent$ADDISON_AKTE
```

Werden diese Befehle eingegeben, kann direkt kopiert oder durch eine Sicherungssoftware ein Backup erstellt werden. Sollen diese Befehle automatisch ausgeführt werden, so besteht die Möglichkeit, die Befehlsfolge als entsprechende Aufgabe in SYSTEMSTEUERUNG | GEPLANTE TASKS einzurichten.

Vergleichbar ist dies mit dem Ausführen des Befehls BEENDEN bzw. STARTEN/WEITER im SQL Server-Dienst-Manager.

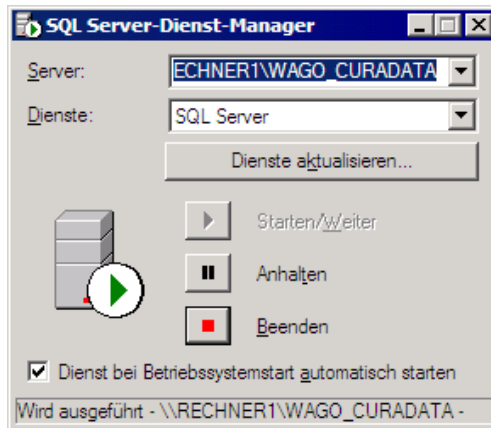


Abbildung 1: SQL Server-Dienst-Manager (MSDE 2000)

Die Methoden 2 und 3 werden von uns für Mehrplatzinstallationen mit vollständigem Wiederherstellungsmodell (Standardeinstellung für Mehrplatzinstallationen, vgl. Kapitel 4.3) als alleinige Methode jedoch nicht empfohlen. Möchten Sie dennoch diese Variante verwenden, richten Sie bei Verwendung dieser Variante Wartungspläne für den SQL Server ein, u.a. um einem übermäßigen Anwachsen des Transaktionsprotokolls vorzubeugen (siehe Kapitel 2.6.3). Sollte es sich aber um eine Einzelplatzinstallation mit einfachem Wiederherstellungsmodell handeln (Standardeinstellung für Einzelplatzinstallationen, vgl. 4.3), dann ist dies nicht notwendig.

Außerdem empfehlen wir, zusätzlich zur tse:nit *banking* Datenbank auch die Datenbanken **MASTER** und **MSDB** zu sichern, damit bei einer Neuinstallation des SQL Servers die tse:nit *banking* Datenbank ohne weitere auszuführende Befehle und Prozeduren zurückgesichert werden kann.

Die msdb-Datenbank enthält zum Beispiel einen vollständigen Verlauf aller Sicherungs- und Wiederherstellungsvorgänge auf dem Server. Der Enterprise Manager verwendet diese Informationen, um einen Wiederherstellungsplan vorzuschlagen und auszuführen, der im Bedarfsfall zum Wiederherstellen einer Datenbank verwendet werden kann. Wenn z.B. jede Nacht eine Datenbanksicherung für eine Benutzerdatenbank erstellt wird und die Sicherungen des Transaktionsprotokolls tagsüber stündlich erstellt werden, werden die im Sicherungsverlauf enthaltenen Informationen in der msdb-Datenbank gespeichert. Muss die Benutzerdatenbank wiederhergestellt werden, kann der SQL Server Enterprise Manager mit Hilfe der in der msdb-Datenbank gespeicherten Verlaufsdaten alle Sicherungen des Transaktionsprotokolls, die sich auf eine bestimmte Datenbanksicherung beziehen, beim Wiederherstellen der Datenbanksicherung anwenden.



Sollten Sie sich für den Einsatz von MSDE 2000 als Datenbank für Ihren Einzelarbeitsplatz entschieden haben, so steht Ihnen die Methode 1 nicht zur Verfügung, da diese Version nicht über den SQL Server Enterprise Manager verfügt. Hier sollten Sie, wie in Methode 3 geschildert, die Dienste stoppen und die entsprechenden Datenbankdateien und das Verzeichnis der Bewegungsdaten sichern.

2.6 Empfohlene Vorgehensweisen zur Sicherung von SQL Server-Datenbanken

In diesem Kapitel schildern wir, wie Sie mit Hilfe des SQL Servers Sicherungen durchführen können. Dabei stellen wir ein sehr einfaches Verfahren vor, um die Datenbank zu sichern, eine mögliche Erweiterung durch einen Zeitplan und schließlich die Einrichtung eines Wartungsplanes.

2.6.1 Manuelle Sicherung mit dem SQL Server Enterprise Manager

Dieses Verfahren beschreibt die Möglichkeit, einfach und schnell eine Sicherung der Datenbank mit Hilfe des Enterprise Managers auszuführen. Diese Vorgehensweise empfiehlt sich vor einem Update oder vor programminternen Anpassungen, die nur über Rücksichern rückgängig gemacht werden können.

Ziel der Sicherung kann entweder eine Datei, die auf der Festplatte abgelegt wird, oder ein Bandlaufwerk sein. Es wird empfohlen, ein anderes Laufwerk (nicht nur andere Partition) als das Laufwerk mit den Originaldaten für diese Datei zu verwenden.

Da der SQL Server das Band in einem eigenen Format beschreibt, wird eine direkte Bandsicherung mit dem SQL Server in den wenigsten Fällen sinnvoll sein. Stattdessen kann die bei der Sicherung entstandene Datei dann wie die anderen zu sichernden Dateien auch im Rahmen der normalen (im allgemeinen Band-) Sicherung gesichert werden.

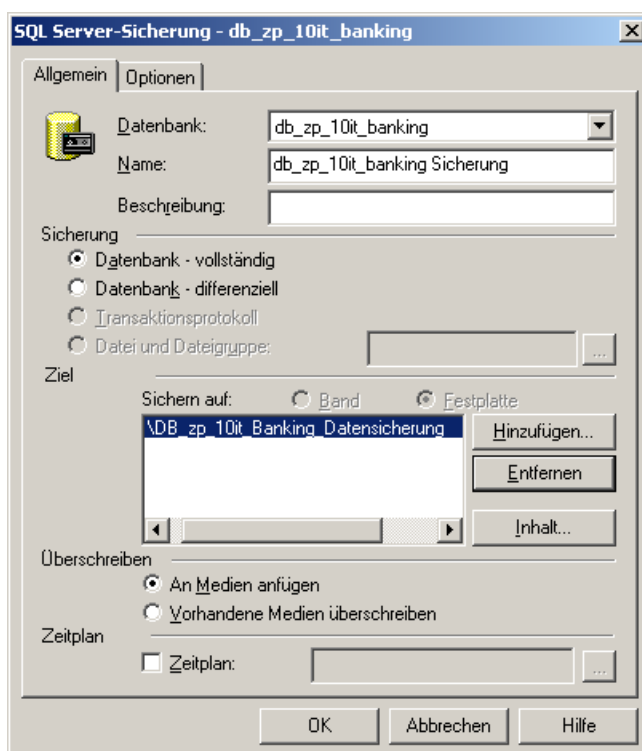


Abbildung 2: SQL Server-Sicherung

Vorgehensweise:

- Auswahl der Datenbank DB_ZP_10IT_BANKING vom Ordner Datenbanken,
- aktivieren des Kontextmenüs mit der rechten Maustaste und Starten des Dialogs DATENBANK SICHERN... unter ALLE TASKS,
- Eingabe eines Namens für die Sicherung,
- optionale Eingabe einer Beschreibung für die Sicherung,
- Auswahl der Sicherungsmethode, im allgemeinen DATENBANK - VOLLSTÄNDIG
- Auswahl oder Hinzufügen des Sicherungsziels (im Beispiel: DB_ZP_10IT_BANKING_DATENSICHERUNG),
- wählen der Option AN MEDIEN ANFÜGEN,
- Start des Sicherungsprozesses mit [OK].



Die Option AN MEDIEN ANFÜGEN setzt voraus, dass entsprechender Platz auf der Festplatte vorhanden ist. Auch sollte die erstellte Sicherungsdatei durch eine externe Sicherungssoftware zum Beispiel auf ein Sicherungsband gesichert werden, wenn diese aufbewahrt werden soll.

2.6.2 Automatisches Sicherungsverfahren

Die Einrichtung eines Terminplanes bzw. Zeitplans erlaubt Ihnen die Einrichtung eines automatischen Sicherungsverfahrens. Der Vorteil besteht darin, dass Sie eine fortlaufende Sicherung erhalten, ohne diese manuell immer wieder zu veranlassen.

Der Nachteil dieser Methode ist, dass bei der Einstellung AN MEDIEN ANFÜGEN ein entsprechend großer Datenträger vorhanden sein muss. Die Einstellung VORHANDENE MEDIEN ÜBERSCHREIBEN hat den Nachteil, dass die vorhergehende Sicherung dann nicht mehr vorhanden ist.

Eine empfehlenswerte Vorgehensweise ist, eine regelmäßige Sicherung mit der Option VORHANDENE MEDIEN ÜBERSCHREIBEN durch einen Zeitplan zu erstellen und dann die erzeugte Sicherungsdatei mit einer speziellen Sicherungssoftware zum Beispiel auf ein Band zu sichern.

Vorgehensweise:

- Schritte wie in Kapitel 2.6.1 beschrieben ausführen.
- Vor dem Sichern mit [OK] ist hierzu die Option ZEITPLAN zu aktivieren. Danach können noch weitere Einstellungen vorgenommen werden.
- Im Dialog ZEITPLAN, Option WIEDERHOLT kann die gewünschte Planung durchgeführt werden, wenn durch Anklicken der Schaltfläche [Ändern] ein entsprechender Zeitplan hinterlegt wird.

Empfohlen wird bei dieser Methode die Durchführung mindestens einer einmal täglichen vollständigen Datenbanksicherung. Bei starkem Arbeiten mit der Datenbank sollten zusätzlich zu der vollständigen Sicherung über den Tag verteilt Transaktionsprotokollsicherungen eingeplant werden. Auch diese können mit diesem Verfahren geplant werden.

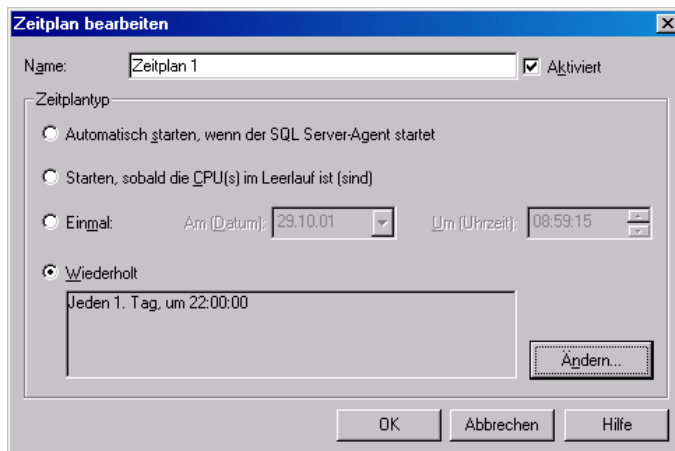


Abbildung 3: Zeitplan bearbeiten



Abbildung 4: Zeitplan für wiederkehrende Aufträge bearbeiten

Damit die angegebenen Termine auch ausgeführt werden, ist der SQL Server-Agent zu starten. Dieser befindet sich unter VERWALTUNG | SQL SERVER-AGENT. Durch Anklicken der rechten Maustaste an dieser Stelle oder auch über den Dienst-Manager kann der Dienst gestartet werden. Ferner sollte der Dienst bei jedem Start des SQL Servers gestartet werden (Kontrollkästchen DIENST BEI BETRIEBSSYSTEMSTART AUTOMATISCH STARTEN im SQL Server-Dienst-Manager aktivieren).

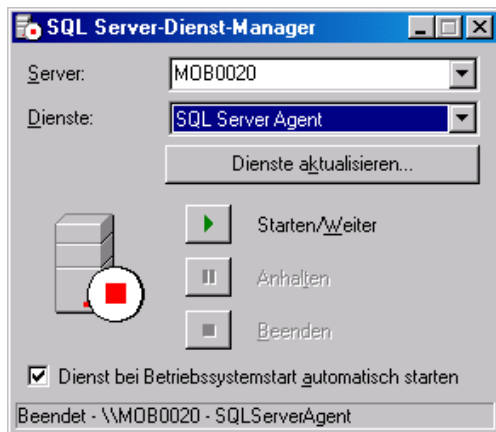


Abbildung 5: SQL Server-Dienst-Manager

Der Nachteil dieses Verfahrens besteht darin, dass diese Sicherungen nicht automatisch gelöscht werden. Es kann zum Nachteil werden, da dies unter Umständen durch den Anwender selbst von Zeit zu Zeit durchgeführt werden muss. Soll dies automatisch geschehen, empfiehlt sich die Einrichtung eines Wartungsplanes.

2.6.3 Datenbank Wartungsplan

2.6.3.1 Voraussetzungen



Um Aufgaben im Rahmen des Wartungsplans auszuführen, ist es erforderlich, dass der SQL Server-Agent-Dienst gestartet ist. Dieser Dienst implementiert den Agenten, der auf der Basis eines Zeitplans Tasks im Rahmen der SQL Server-Administration ausführt. Für jede Instanz von SQL Server, die auf einem Computer ausgeführt wird, gibt es einen eigenen SQL Server-Agent-Dienst.

Bitte konfigurieren Sie den Dienst so, dass der SQL Server den Dienst automatisch startet. Der Status der ausgeführten Wartungspläne muss regelmäßig kontrolliert werden. Der SQL Server bietet hierzu diverse Unterstützungsdienste an, z.B. das automatisierte Versenden von E-Mails.

Benutzen Sie zur Konfiguration des SQL Server-Agenten den Bereich VERWALTUNG | SQL SERVER-AGENT im Enterprise Manager (siehe Abbildung 6).



Abbildung 6: Aufträge

2.6.3.2 Aufruf des Datenbank-Wartungsplanungs-Assistenten

Mit Hilfe des Datenbank-Wartungsplanungs-Assistenten können Sie eine Reihe von zentralen Wartungstasks einrichten, die notwendig sind, um die Leistungsfähigkeit der Datenbank, die regelmäßige Sicherung der Datenbank zur Absicherung bei Systemausfällen und die Überprüfung auf Inkonsistenzen in der Datenbank sicherzustellen. Der Datenbank-Wartungsplanungs-Assistent erstellt einen Microsoft® SQL Server™-Auftrag, der diese Wartungstasks automatisch in geplanten Intervallen ausführt.

Mit Hilfe des Assistenten legen Sie Ihren Wartungsplan an. Dabei legen Sie die gewünschten Einstellungen fest. Rufen Sie den Assistenten über das Kontextmenü VERWALTUNG | DATENBANK-WARTUNGSPLAN durch Auswahl von NEUER WARTUNGSPLAN... auf.

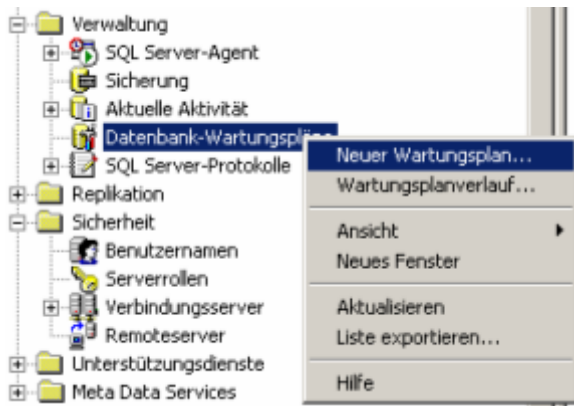


Abbildung 7: Datenbank-Wartungspläne Kontextmenü

Nachträglich kann dieser Wartungsplan angepasst werden, indem Sie die Eigenschaften (Rechtsklick auf Objekt DB-Wartungsplan) bearbeiten.

2.6.3.3 Datenbanken auswählen

Wählen Sie hier die gewünschte Datenbank, für die ein Wartungsplan erstellt werden soll, z.B. die Datenbank DB_ZP_10IT_BANKING. Wenn Sie auch die Systemdatenbanken im Rahmen eines Wartungsplanes sichern möchten, nehmen Sie diese nicht in den Wartungsplan für die tse:nit *banking*-Datenbank(en) auf, sondern verwenden Sie einen gesonderten Wartungsplan.

2.6.3.4 Optimierungen im Datenbank-Wartungsplan

Verwenden Sie die Seite PARAMETER FÜR DIE DATENOPTIMIERUNG AKTUALISIEREN des Assistenten bzw. die Registerkarte OPTIMIERUNGEN, um Daten- und Indexseiten neu zu organisieren und damit dem Abfrageoptimierer besseren Zugriff auf Ausführungspläne zu ermöglichen.

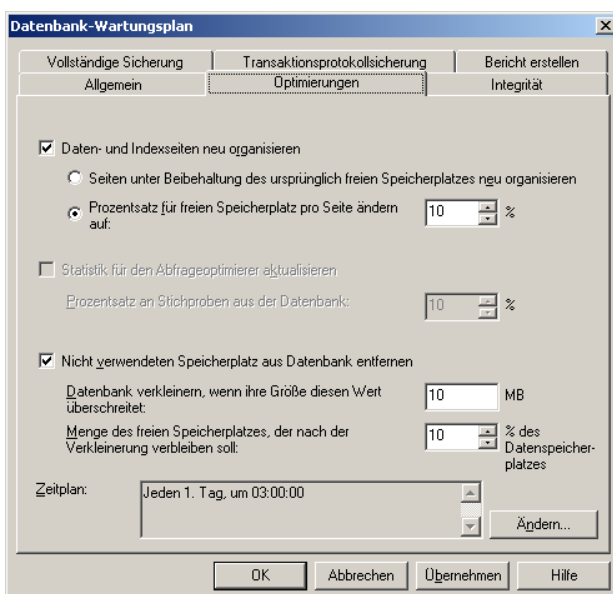


Abbildung 8: Datenbank-Wartungsplan | Registerblatt Optimierungen

2.6.3.5 Datenintegritätstests im Datenbank-Wartungsplan

Verwenden Sie die Seite DATENBANKINTEGRITÄTSPRÜFUNG im Assistenten bzw. die Registerkarte INTEGRITÄT, um Datenintegritätstests auszuführen, durch die sich beschädigte Daten erkennen lassen. Eine Beschädigung kann unter anderem durch Hardwarefehler oder Fehler der Systemsoftware verursacht werden.

Die Option KLEINERE PROBLEME BEHEBEN darf **nicht** aktiviert werden, wenn im Rahmen des Wartungsplans die Datenbanken MASTER und/oder MSDB überprüft werden. Sie erhalten sonst unter Umständen folgende Fehlermeldung:

REPAIR-Anweisung nicht verarbeitet. Datenbank muss sich im Einzelbenutzermodus befinden.

Legen Sie gegebenenfalls einen separaten Wartungsplan an, um diese Datenbanken zu überprüfen. Für die Überprüfung der tse:nit Datenbank sollte die Option KLEINERE PROBLEME BEHEBEN aktiviert werden.

Wenn Sie Datenbankintegritätstest durch den Wartungsplan durchführen lassen, prüfen Sie bitte regelmäßig die entstandenen Berichte und das SQL Server-Protokoll bzw. richten Sie Warnungen mit entsprechenden Benachrichtigungen ein. Weitergehende Informationen entnehmen Sie bitte der SQL Server-Onlinedokumentation.

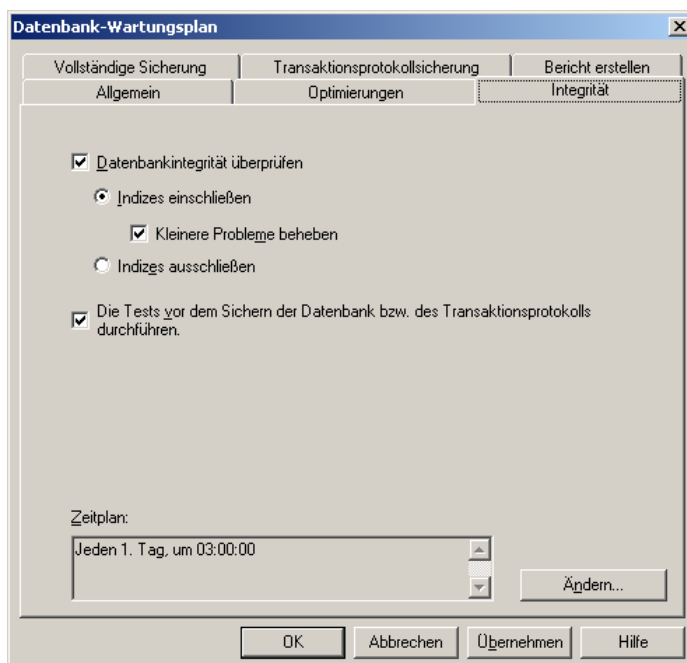


Abbildung 9: Datenbank-Wartungsplan Registerblatt Integrität

2.6.3.6 Sicherung der Datenbank

Auf den Seiten SICHERUNGSPLAN FÜR DATENBANK FESTLEGEN und FESTPLATTENVERZEICHNIS FÜR SICHERUNG ANGEBEN des Assistenten bzw. auf der Registerkarte VOLLSTÄNDIGE SICHERUNG legen Sie die Parameter für die vollständige Sicherung der Datenbank fest. Über die Schaltfläche [Ändern] können Sie bestimmen, wie häufig gesichert werden soll. Außerdem ist es möglich, Dateien die älter sind als zum Beispiel eine Woche, automatisch zu löschen.

Beachten Sie auch hier, dass eine Sicherung auf der Festplatte nicht ausreicht, wenn es zu einem Verlust dieser kommt.

2.6.3.7 Transaktionsprotokollsicherung - Abschneiden des Transaktionsprotokolls

Durch Transaktionsprotokollsicherungen werden Sie in die Lage versetzt, den Status der Datenbank zu einem bestimmten Zeitpunkt (z.B. bis zu dem Zeitpunkt, bevor unerwünschte Daten eingegeben wurden) oder bis zu dem Punkt, an dem ein Fehler aufgetreten ist, wiederherzustellen. Außerdem wird durch Transaktionsprotokollsicherungen das Transaktionsprotokoll abgeschnitten.

Das Transaktionsprotokoll ist eine fortlaufende Aufzeichnung aller Transaktionen, die in der Microsoft® SQL Server™-Datenbank ausgeführt wurden, seit das Transaktionsprotokoll zuletzt gesichert wurde.

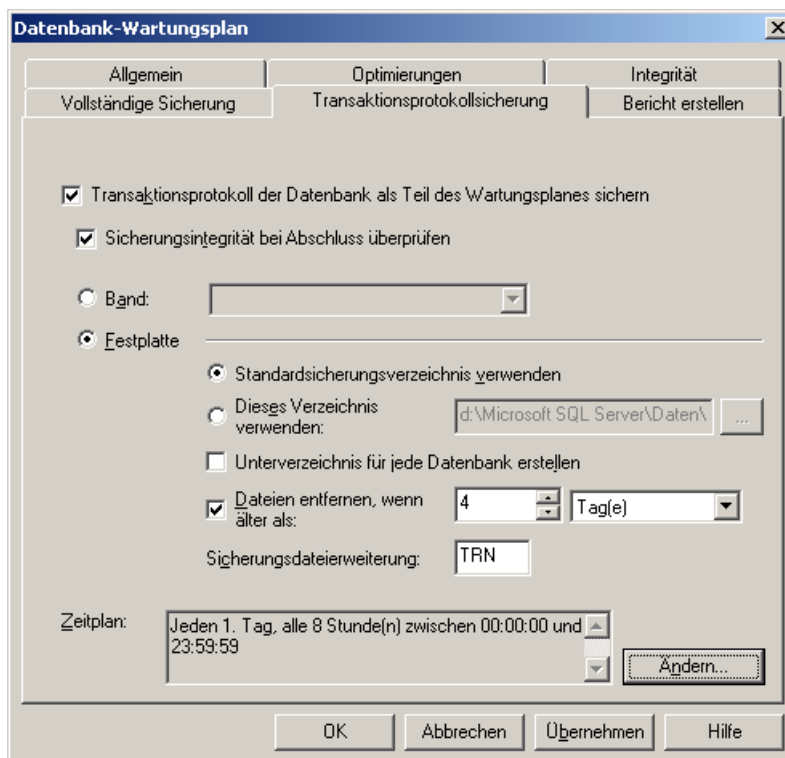


Abbildung 10: Datenbank-Wartungsplan Registerblatt Transaktionsprotokollsicherung

Erfolgt keine Löschung der Protokolleinträge aus dem Transaktionsprotokoll, würde das logische Protokoll so lange wachsen, bis es den gesamten verfügbaren Speicherplatz einnimmt, der auf dem Datenträger vorhanden ist, auf dem sich die physischen Protokolldateien befinden. Von Zeit zu Zeit müssen alte Protokolleinträge, die zum Wiederherstellen einer Datenbank nicht mehr benötigt werden, gelöscht werden, um Speicherplatz für neue Protokolleinträge freizugeben. Der Vorgang, durch den diese Protokolleinträge gelöscht werden, um die Größe des logischen Protokolls zu verringern, wird als Abschneiden des Protokolls oder Protokollkürzung bezeichnet.



SQL Server schneidet das Transaktionsprotokoll nur bei einer Transaktionsprotokollsicherung ab, jedoch nicht, wenn eine vollständige oder differenzielle Sicherung der Datenbank durchgeführt wird.

Führen Sie zusätzlich zur täglichen vollständigen Datensicherung ein- oder mehrmals täglich eine Transaktionsprotokollsicherung durch. Legen Sie hierzu einen Datenbank-Wartungsplan an und aktivieren Sie auf der Seite SICHERUNGSPLAN FÜR TRANSAKTIONSPROTOKOLL FESTLEGEN des Assistenten bzw. auf der Registerkarte TRANSAKTIONSPROTOKOLLSICHERUNG die Auswahl TRANSAKTIONSPROTOKOLL ALS TEIL DES WARTUNGSPLANES SICHERN (siehe Abbildung 10).

Transaktionsprotokollsicherungen werden nur mit dem Modell der vollständigen Wiederherstellung bzw. dem Modell der massenprotokollierten Wiederherstellung verwendet. Es kommt vor, dass eine Transaktionsprotokollsicherung umfangreicher ist als eine Datenbanksicherung, beispielsweise wenn eine Datenbank eine hohe Transaktionsrate aufweist, wodurch sich das Transaktionsprotokoll schnell vergrößert. In dieser Situation sollten Sie häufiger Sicherungen des Transaktionsprotokolls erstellen.

Das Wiederherstellen einer Datenbank mit Hilfe von Datenbank- und Transaktionsprotokollsicherungen funktioniert nur bei der Verwendung einer ununterbrochenen Sequenz von Transaktionsprotokollsicherungen nach der letzten Datenbanksicherung oder der letzten differenziellen Datenbanksicherung.



Führen Sie deshalb regelmäßig vollständige Datensicherungen durch. Empfohlen wird die tägliche vollständige Datensicherung (in der Regel nachts vor Durchführung der Optimierungen und Integritätsprüfungen).

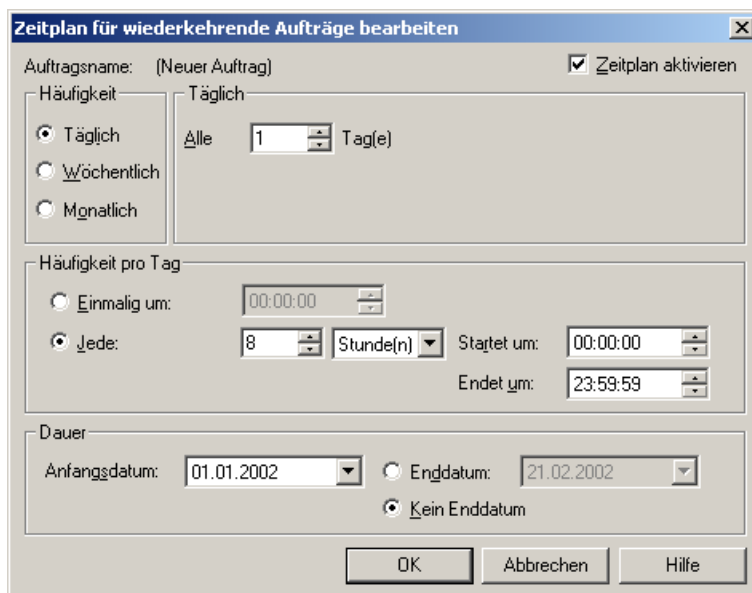


Abbildung 11: Zeitplan für wiederkehrende Aufträge bearbeiten

Wenn der SQL Server das Sichern des Transaktionsprotokolls beendet, wird der inaktive Abschnitt des Transaktionsprotokolls automatisch abgeschnitten.

Aktivieren Sie die Option DATEIEN ENTFERNEN, WENN ÄLTER ALS, um automatisch Transaktionsprotokollsicherungen, die ein bestimmtes Alter überschritten haben, zu löschen. Hierdurch wird der verwendete Festplattenspeicher minimiert. Es muss jedoch sichergestellt sein, dass jederzeit die benötigten Datenbank- und Transaktionsprotokollsicherungen (z.B. auf einem Bandmedium) zur Verfügung stehen. Die Sicherung auf der Festplatte ist nicht ausreichend, wenn es zu einem Verlust dieser kommt.

Beinhaltet der Wartungsplan neben der tse:nit Datenbank noch weitere (z.B. die master-Datenbank), muss sichergestellt sein, dass für alle Datenbanken das VOLLSTÄNDIGE WIEDERHERSTELLUNGSMODELL aktiviert ist. Ansonsten ist das automatische Löschen alter Transaktionsprotokollsicherungen nicht sichergestellt. Legen Sie in diesem Fall mehrere Wartungspläne an.

2.6.3.8 Textberichte / Verlauf

Verwenden Sie Textberichte und/oder Verlaufsinfos, um einen Bericht zu den von Microsoft® SQL Server™ ausgeführten Wartungsaktivitäten in einer Datei zu speichern bzw. in eine Tabelle zu schreiben.

2.6.3.9 Wartungsplan und Systemdatenbanken

Wenn Sie einen Wartungsplan für die tse:nit *banking*-Datenbank einrichten, müssen Sie darauf achten, dass die Systemdatenbanken MASTER und MSDB folgende Voraussetzungen erfüllen:

- Stellen Sie für die Systemdatenbanken MASTER und MSDB grundsätzlich das einfache Wiederherstellungsmodell ein.
- Richten Sie für die Systemdatenbanken einen eigenen Wartungsplan ein.
- Führen Sie für die Systemdatenbanken keine Transaktionsprotokollsicherung durch. Dies ist beim einfachen Wiederherstellungsmodell nicht möglich.

Wenn diese Voraussetzungen nicht eingehalten werden, kann dies zu nicht vollständig ausgeführten Wartungsplänen führen. Auch Ihre im Rahmen eines Wartungsplanes eingerichtete Sicherung des Transaktionsprotokolls der tse:nit *banking*-Datenbank wird nicht ordnungsgemäß ausgeführt. In der Folge wird Ihr Transaktionsprotokoll nicht mehr verkleinert und wächst überproportional an.

3 Vorgehensweise zur Rücksicherung der Datenbank

3.1 Rücksicherung der Datenbanksicherungsdatei

Haben Sie eine Datenbanksicherung mit dem Enterprise Manager Ihres SQL Servers durchgeführt, müssen Sie eine Rücksicherung wie in diesem Kapitel geschildert durchführen. Als Beispiel nehmen wir eine tse:nit *banking*-Datenbank.



Beachten Sie bitte, dass Sie Ihre Bewegungsdaten auf den entsprechenden Stand der Datenbank halten. Kopieren Sie die gesicherten Bewegungsdaten in das Verzeichnis 10IT_BANKING_DATEN (Standardname) bzw. in Ihr Programmverzeichnis (bei älteren Einzelplatzinstallationen), aber nicht ohne vorher vom letzten Stand ebenfalls eine Sicherung erstellt zu haben.

Ist die tse:nit *banking*-Datenbank im Enterprise Manager als Objekt vorhanden, rufen Sie das Kontextmenü auf und wählen ALLE TASKS | DATENBANK WIEDERHERSTELLEN... aus. In dem folgenden Dialog können Sie aus allen Ihren Sicherungen die Richtige auswählen (siehe Abbildung 12).

Vorgehensweise:

1. Im Feld WIEDERHERSTELLEN ALS DATENBANK wählen Sie die Datenbank aus. Da Sie von der vorhandenen tse:nit Datenbank den Befehl aufgerufen haben, ist dies Ihr Standardvorschlag. Wenn Sie eine Sicherung testen wollen, könnten Sie hier einen abweichenden Namen eingeben.
2. Wählen Sie WIEDERHERSTELLEN als Datenbank aus.
3. Lassen Sie sich Sicherungskopien der DB_ZP_10IT_BANKING anzeigen.
4. Im Feld ERSTE WIEDERHERZUSTELLENDEN SICHERUNG können Sie das Medium, also Ihre Sicherungsdatei, auswählen.
5. Es werden alle vorhandenen Sicherungen innerhalb des ausgewählten Mediums aufgelistet.
6. Durch Ankreuzen wählen Sie die Sicherung aus. In unserem Beispiel handelt es sich um ein Transaktionsprotokoll, das etwas eingerückt dargestellt wird. Automatisch ist die letzte vollständige Sicherung mit angekreuzt. Wenn Sie die Schaltfläche [Eigenschaften] anklicken, erhalten Sie weitere Informationen zur markierten Sicherung.
7. Mit [OK] startet die Rücksicherung.

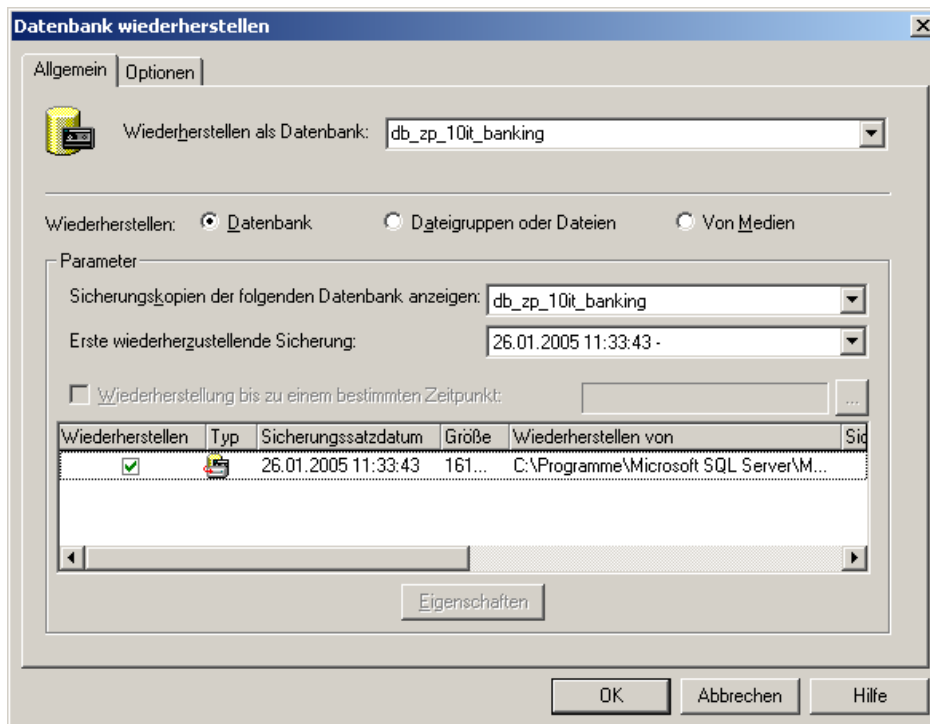


Abbildung 12: Datenbank wiederherstellen

Wenn Sie über eine Sicherung verfügen, aber die Datenbank DB_10IT selbst im Enterprise Manager nicht mehr angezeigt wird, haben Sie die Möglichkeit, die Sicherung einzuspielen indem Sie das Kontextmenü des Objektes DATENBANKEN aufrufen und über **Alle TASKS | DATENBANK WIEDERHERSTELLEN** diesen Dialog öffnen. Voraussetzung ist, dass die Systemdatenbanken MASTER und MSDB über die entsprechenden Informationen verfügen.

Ist dies nicht der Fall, weil Sie diese Systemdatenbanken nicht gesichert haben und Sie nur über die tse:nit Datenbanksicherung verfügen, dann müssen Sie folgendermaßen vorgehen (siehe Abbildung 13):

1. Sicherung auf Festplatte kopieren,
2. Schreibschutz entfernen, falls die Sicherung von einer CD stammt,
3. Befehl **ALLE AUFGABEN | DATENBANK WIEDERHERSTELLEN** auf Objekt Datenbanken im Enterprise Manager aufrufen,
4. Datenbankname eingeben,
5. Option **VON MEDIEN** auswählen,
6. Befehl **MEDIEN AUSWÄHLEN**,
7. Befehl **HINZUFÜGEN**,
8. Sicherungsdatei auswählen,
9. [OK].

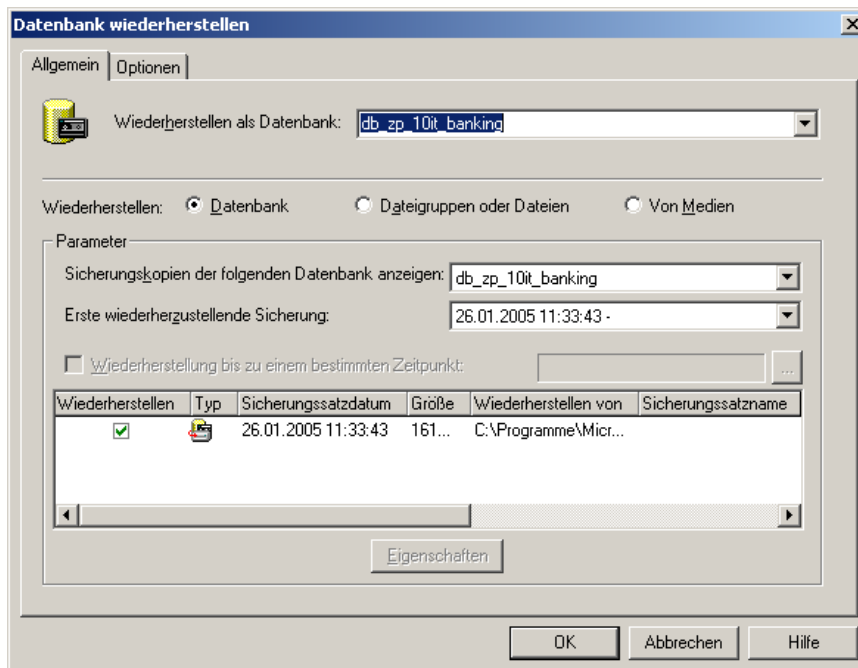


Abbildung 13: Datenbank wiederherstellen - von Medien

Stammt die Sicherung nicht aus der gleichen SQL Server-Installation (also einem anderen Rechner oder eine neue SQL Server-Installation auf gleichem Rechner), muss der für tse:nit *banking* benötigte Benutzer `BASQLDBSERVER` der Datenbank dem Benutzer `BASQLDBSERVER` des SQL Servers wieder zugeordnet werden.

Handelt es sich um eine SQL Server-Neuinstallation, muss zuvor der Benutzer `BASQLDBSERVER` noch im SQL Server selbst angelegt werden. Führen Sie dazu in den tse:nit *banking administration tools* unter **ALLGEMEINE AUFGABEN** die Aufgabe **SQL SERVER-ERSTEINRICHTUNGS-DATENBANK ANLEGEN** aus und benennen die Datenbank mit `DB_ZP_10IT_BANKING_ERST`. Öffnen Sie im Anschluss das Programm Query Analyzer, welches Bestandteil des SQL Servers ist.

Geben Sie folgende Befehle ein und führen diese jeweils aus:

1. Befehl eingeben:
use db_zp_10it_banking
 Führen Sie Menü **ABFRAGE | AUSFÜHREN** aus.
2. Befehl eingeben:
EXEC sp_change_users_login Update_One, 'BASQLDBServer', 'BASQLDBServer'
 Führen Sie Menü **ABFRAGE | AUSFÜHREN** aus.

Danach können Sie auf die zurückgesicherte Datenbank mit den tse:nit *banking* zugreifen.

3.2 Rücksicherung der Datenbankdatei und des Transaktionsprotokolls

Haben Sie die Datenbankdateien DB_ZP_10IT_BANKING.MDF und das Transaktionsprotokoll DB_ZP_10IT_BANKING_LOG.LDF gesichert, können Sie die nachfolgend beschriebenen Vorgehensweisen zur Rücksicherung durchführen.



Beachten Sie bitte, dass sie Ihre Bewegungsdaten auf den entsprechenden Stand der Datenbank halten. Kopieren Sie die gesicherten Bewegungsdaten in das Verzeichnis 10it_banking_Daten (Standardname) bzw. in Ihr Programmverzeichnis (bei älteren Einzelplatzinstallationen), aber nicht ohne vorher vom letzten Stand ebenfalls eine Sicherung erstellt zu haben.

Vorgehensweise 1:

1. Beenden Sie die Dienste des SQL Servers (siehe Kapitel 3.1).
2. Machen Sie eine Sicherung der Dateien, die Sie überschreiben wollen.
3. Kopieren Sie Ihre gesicherten Dateien in das Verzeichnis DATA (siehe Kapitel 3.1).
4. Wenn Sie die Datenbankdateien der Datenbanken MSDB und MASTER ebenfalls gesichert haben, können Sie auch diese einspielen.
5. Starten Sie die Dienste des SQL Servers (siehe Kapitel 3.1).

Vorgehensweise 2:

Wenn Sie über einen Enterprise Manager (gilt also nicht für MSDE 2000) verfügen, können Sie über dieses Programm eine Datenbank anhängen (siehe Abbildung 14).

1. Im Enterprise Manager öffnen Sie das Kontextmenü des Ordners Datenbanken.
2. Wählen Sie den Befehl ALLE TASKS | DATENBANK ANHÄNGEN... aus.
3. Dann wählen Sie die Datenbankdatei aus, die sich im Verzeichnis DATA Ihres SQL Servers befindetet.
4. Wenn vorher noch keine tse:nit *banking*-Datenbank vorhanden war, müssen Sie im Query Analyzer den verwaisten Benutzer reparieren (siehe Kapitel 3.1).

Der korrespondierende Befehl, eine Datenbank auszuhängen, kann über das Kontextmenü der Datenbank (Objekt DB_ZP_10IT_BANKING) ALLE TASKS | DATENBANK TRENNEN... ausgeführt werden.



Wichtig ist bei allen Vorgehensweisen, dass Sie mit größtmöglicher Sicherheit arbeiten. Erstellen Sie zum Beispiel jeweils vom letzten Stand Ihres Datenbestandes eine Sicherung bzw. kopieren Sie die Bewegungsdaten und die Datenbank, auch wenn Sie glauben, diese nicht mehr zu benötigen.

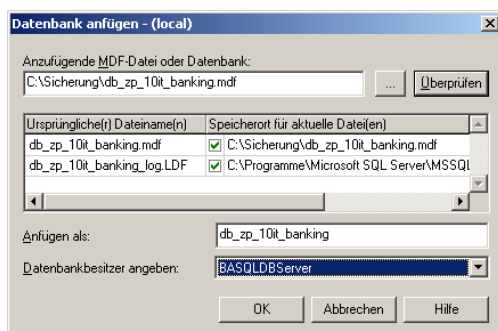


Abbildung 14: Datenbank anfügen (SQL Server 2000)

4 Hinweise zum Einsatz von tse:nit mit Microsoft® SQL Server™

4.1 Voraussetzungen

Um eine Datensicherung durchzuführen oder die nachfolgend beschriebenen Einstellungen vorzunehmen, benötigen Sie den SQL Server Enterprise Manager. Der Enterprise Manager ist das primäre Administrationstool für Microsoft® SQL Server™ und ist Bestandteil des SQL Server.

Nachfolgend werden weitere wichtige Voraussetzungen genannt.



Grundsätzlich ist vor dem Durchführen von Änderungen am SQL Server eine Sicherung der tse:nit *banking*-Datenbank zu erstellen.



Sollten Sie sich für den Einsatz von MSDE 2000 als Datenbank entschieden haben, so stehen Ihnen die nachfolgend genannten Möglichkeiten nicht zur Verfügung, da diese Version nicht über SQL Server Enterprise Manager verfügt.

4.1.1 Service-Pack-Versionen

Installieren Sie nach dem SQL Server auch das aktuell verfügbare Service Pack (wird durch die ADDISON Software und Service GmbH mitgeliefert oder ist im Internet bei Microsoft verfügbar). Anbei eine Übersicht der verfügbaren Service Packs:

Version	SQL Server 2000-Version
8.00.194	SQL Server 2000 RTM
8.00.384	Service Pack 1
8.00.532 bzw. 8.00.534	Service Pack 2
8.00.760	Service Pack 3 bzw. 3a

Tabelle 1: Service Pack-Versionen SQL Server 2000-Datenbankmoduls

Für den Einsatz des SQL Servers werden bestimmte Service Packs vorausgesetzt. Entsprechende Informationen hierzu finden Sie in dem Dokument SYSTEMVORAUSSETZUNGEN 10IT. Sie finden die aktuellen Service Packs auf der tse:nit CD-ROM oder können diese aus dem Internet von Microsoft herunterladen.



Über welchen Service Pack Ihr SQL Server verfügt, können Sie mit Hilfe des Tasks ANALYSE | DATENBANK -INFORMATIONEN | AUSFÜHREN... im Programm tse:nit *administration tools* feststellen. Oder entnehmen Sie diese Information dem Enterprise Manager, indem Sie im Kontextmenü des Objektes Server den Befehl Info aufrufen.

4.1.2 Administrator-Passwort

Bitte achten Sie darauf, dass bei Ihrem SQL Server unbedingt ein Passwort für den Administrator-Zugang (Datenbankbenutzer sa) vergeben wird. Wie *heise online* berichtete, besteht ansonsten die Gefahr eines Befalls durch Computerviren. Fragen Sie gegebenenfalls Ihren Systembetreuer, ob er für den Benutzer sa ein Kennwort vergeben hat.

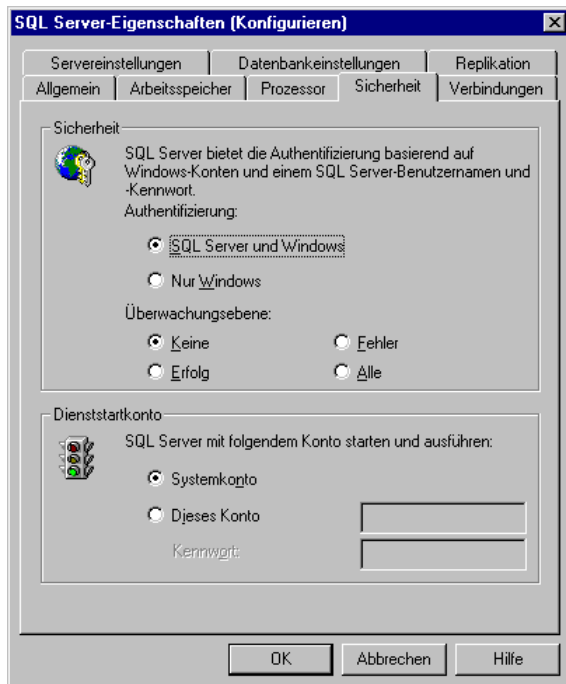


Abbildung 15: SQL Server-Eigenschaften (Konfigurieren) - Sicherheit

Dieses Passwort sollte zur Administrierung der Datenbank bekannt sein. Bei der von uns empfohlenen gemischten Authentifizierung besteht weiterhin die Möglichkeit, sich mit einem Windows Benutzer Konto anzumelden. Diese Eigenschaft des SQL Servers kann im Enterprise Manager (Rechtsklick auf Objekt SERVER | EIGENSCHAFTEN | Registerblatt SICHERHEIT (siehe Abbildung 15) eingestellt werden.

4.1.3 Enterprise Manager

Der Enterprise Manager ist ein Programm des SQL Servers, das Ihnen Möglichkeiten zur Wartung Ihrer tse:nit *banking* Datenbank zur Verfügung stellt. Wenn Sie dieses Programm über START | PROGRAMME | MICROSOFT SQL SERVER starten, müssen Sie sich zunächst registrieren.

4.2 Eigenschaften der Datenbank

Starten Sie den Enterprise Manager und öffnen Sie die Eigenschaftendialog für Ihre Datenbank.



Abbildung 16: Kontextmenü Datenbank

Die erforderlichen Einstellungen werden in der Regel auf dem Registerblatt OPTIONEN vorgenommen:

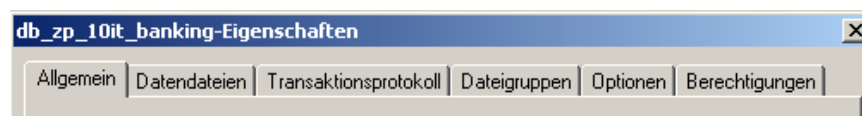


Abbildung 17: db_zp_10it_Banking-Eigenschaften

Das Kontrollkästchen ERKENNUNG VON ZERRISSENEN SEITEN legt fest, dass unvollständige Seiten erkannt werden. Diese Wiederherstellungsoption ermöglicht, dass SQL Server unvollständige Ein-/Ausgabe-Operationen erkennt, die durch Strom- oder Systemausfälle verursacht wurden. Wenn Sie durch Batterien abgesicherte Datenträgercaches verwenden, können Sie sicherstellen, dass Daten entweder erfolgreich oder gar nicht auf den Datenträger geschrieben werden.



Sollten sich durch Einschalten dieser Optionen Performance-Verschlechterungen ergeben, sollte die Option testweise abgeschaltet werden. Es sind dann zwingend Datenbankwartungspläne einzurichten, da nach der Erkennung von zerrissenen Seiten unter Umständen eine Datenbank als fehlerverdächtig gekennzeichnet wird und nicht mehr verwendbar ist, so dass auf eine Datensicherung zurückgegriffen werden muss.



Das Kontrollkästchen REKURSIVE TRIGGER muss aktiviert sein. Ist diese Option nicht aktiviert, sind schwerwiegende Fehler in der Datenbank nicht auszuschließen, die eine Weiterarbeit mit tse:nit unmöglich machen.

4.3 Wiederherstellungsmodell

Sie können für jede Datenbank in Microsoft® SQL Server™ 2000 unter drei Wiederherstellungsmodellen auswählen, um zu bestimmen, wie die Daten gesichert werden sollen und welches Risiko des Datenverlustes Sie eingehen möchten. Die folgenden Wiederherstellungsmodelle sind verfügbar:

- **Einfache Wiederherstellung**
 Mit der einfachen Wiederherstellung kann die Datenbank bis zur letzten vollständigen Sicherung der Datenbank wiederhergestellt werden. Es wird keine Folge von Transaktionsprotokollen gesichert. Eine einfache Wiederherstellung kann empfohlen werden, wenn es sich zum Beispiel um eine Einzelplatzinstallation mit geringen Veränderungen des Datenbestandes handelt oder nur wenig freier Speicherplatz auf der Festplatte zur Verfügung steht.

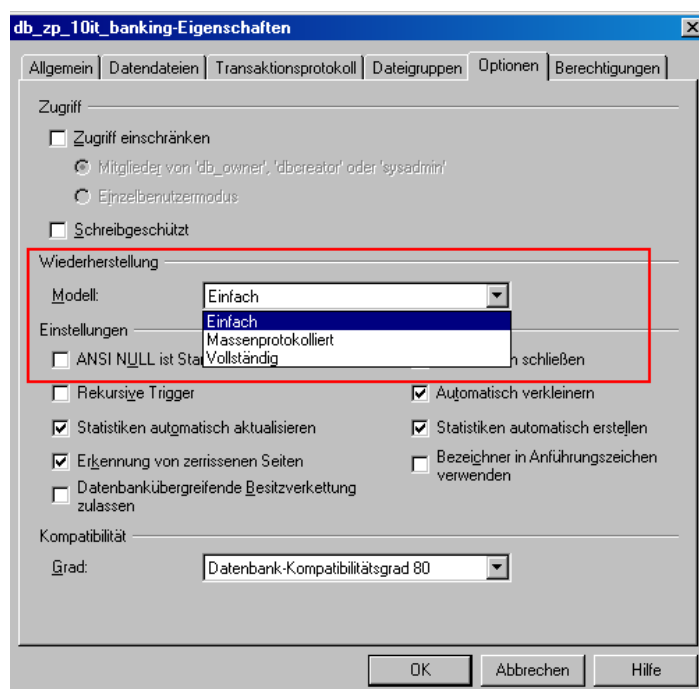


Abbildung 18: db_zp_10it_Banking-Eigenschaften | Registerblatt Optionen

- **Vollständige Wiederherstellung**
 Mit der vollständigen Wiederherstellung kann die Datenbank bis zum Auftreten des Fehlers wiederhergestellt werden.

Beim Verlust einer Datenbank können Sie die Daten wiederherstellen, indem Sie die letzte Datenbanksicherung und anschließend jede Protokollsicherung wiederherstellen, die seit der Datenbanksicherung erstellt wurde. Hierzu muss die Folge von Protokollsicherungen jeden Protokolleintrag enthalten, der seit der letzten Datenbanksicherung geschrieben wurde. Wenn Sie eine Folge von Transaktionsprotokollsicherungen verwalten, darf ein Protokolleintrag erst dann abgeschnitten werden, wenn er in eine Protokollsicherung geschrieben worden ist.

Wir empfehlen, bei der Option WIEDERHERSTELLUNG | MODELL die Auswahl VOLLSTÄNDIG zu wählen. Die im Folgenden beschriebene Transaktionsprotokollsicherung ist nur bei dieser Einstellung möglich. In tse:nit banking Datenbanken ist dieses Wiederherstellungsmodell standardmäßig eingestellt.

- **Massenprotokollierte Wiederherstellung**
 Das Modell der massenprotokollierten Wiederherstellung bietet Schutz vor Medienfehlern und zugleich

die beste Leistung und den geringsten Verbrauch an Protokollspeicherplatz für bestimmte umfangreiche Vorgänge oder Massenkopiervorgänge.

Beim Modell der massenprotokollierten Wiederherstellung ist die Gefahr des Datenverlustes bei diesen Massenkopiervorgängen höher als beim Modell der vollständigen Wiederherstellung. Eine beschädigte Datendatei kann dazu führen, dass Daten erneut manuell eingegeben werden müssen.

4.4 Verkleinern der Datenbankdateien beim SQL Server

Der SQL Server ermöglicht, dass jede Datei innerhalb einer Datenbank verkleinert wird, um nicht verwendete Speicherbereiche zu entfernen. Sowohl Daten- als auch Transaktionsprotokolldateien können dabei verkleinert werden.

Ist also z.B. die Festplatte mit den SQL Server-Datenbanken voll und ist im Enterprise Manager (auf der Seite RESERVIERTER SPEICHER) oder im Explorer (Datei DB_10IT_LOG.LDF) zu sehen, dass das Transaktionsprotokoll viel zu groß ist (mehrere Hundert Megabyte) sind die folgende Aktionen durchzuführen.

4.4.1 Datenbank verkleinern

Im Enterprise Manager über rechte Maustaste das Kontextmenü der Datenbank aufrufen, ALLE TASKS (bzw. ALLE AUFGABEN) und dann DATENBANK VERKLEINERN auswählen.

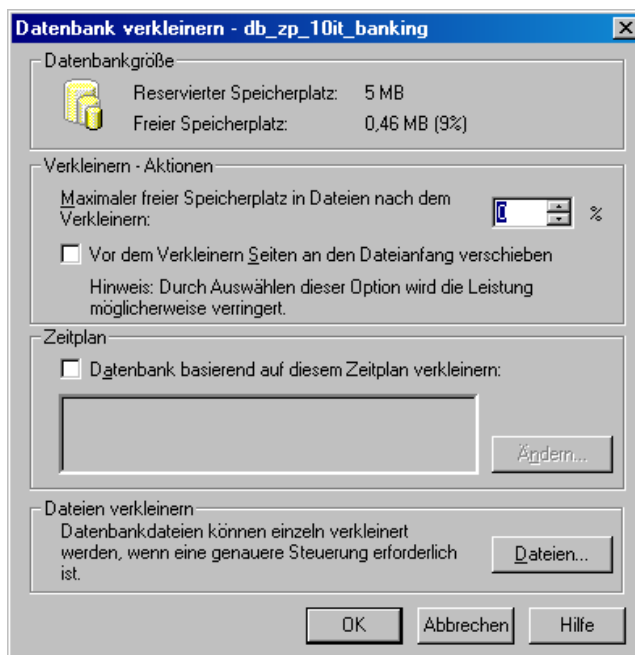


Abbildung 19: Datenbank verkleinern - db_zp_10it_Banking

Das Verkleinern einer Transaktionsprotokolldatei führt nicht zum sofortigen Verkleinern der Datei; stattdessen wird die Datei für die spätere Verkleinerung beim nächsten Abschneiden des Transaktionsprotokolls gekennzeichnet.

4.4.2 Weiteres Vorgehen

Sollte diese Vorgehensweise bezüglich des Transaktionsprotokolls noch nicht zum gewünschten Erfolg geführt haben, kann über folgende Vorgehensweise ein kleines, leeres Transaktionsprotokoll angelegt werden:

1. Datenbank sichern.
2. Sicherstellen, dass freie Festplattenkapazität vorhanden ist (mind. 100 MB für die nachfolgenden Aktionen).
3. Im Enterprise Manager das Kontextmenü der Datenbank aufrufen, ALLE AUFGABEN (bzw. ALLE TASKS) und dann DATENBANK TRENNEN... auswählen.
4. Im Explorer die Datei DB_ZP_10IT_BANKING_LOG.LDF (bei Standardinstallation im Verzeichnis C:\Programme\Microsoft SQL Server\80\MSSQL\data) umbenennen.
5. Das Kontextmenü des Ordners DATENBANKEN aufrufen, ALLE AUFGABEN (bzw. ALLE TASKS) und dann DATENBANK ANHÄNGEN... auswählen.
6. Nach Auswahl der Datenbankdatei DB_ZP_10IT_BANKING.MDF (Schaltfläche [...]) und [OK] wird die Datenbank an den SQL Server angehängt und ein neues Transaktionsprotokoll erstellt.
7. Wenn die Datenbank verbunden und das Transaktionsprotokoll neu angelegt wurde, kann die alte umbenannte LOG Datei gelöscht werden.



Diese Vorgehensweise darf nur als allerletzte Möglichkeit gesehen werden, die LOG-Datei zu verkleinern. Keinesfalls sollte diese Vorgehensweise routinemäßig eingesetzt werden. Bei einem entsprechend eingerichteten Wartungsplan ist eine derartige Vorgehensweise unnötig.

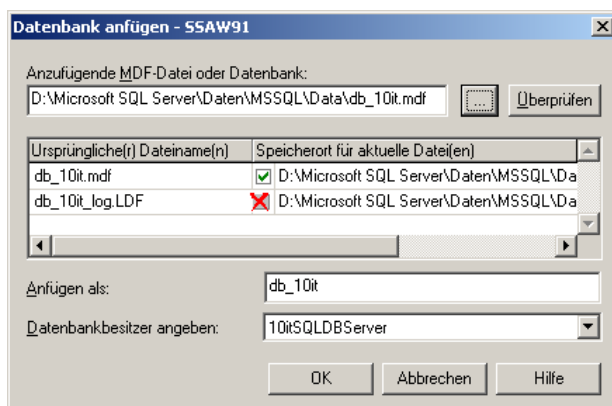


Abbildung 20: Datenbank anfügen

5 Performancesteigerung

Dieses Kapitel schildert Einstellungen und Maßnahmen, die die Performance Ihrer tse:nit Installation verbessern. Es handelt sich hierbei jedoch nicht Vorgaben, die eingehalten werden müssen.

5.1 Voraussetzungen

5.1.1 SQL Server

1. Installation des SQL Server 2000 inkl. des Client-Tools ENTERPRISE MANAGER.
2. Installation des aktuellen Service Packs des jeweiligen SQL Servers, z.Z. Service Pack 3a für den SQL Server 2000.
3. Bezüglich der Hardware ist den einschlägigen Systemvoraussetzungen Genüge zu leisten. Je nach Anzahl der Benutzer bedeutet dies insbesondere, falls möglich:
 - Mehrfachprozessor inkl. Prozessorcaché,
 - großer Hauptspeicher,
 - schnelles Bus-System,
 - mehrere Festplatten mit eigenen Controllern.

5.1.2 Betriebssystem

Bei den laut Systemvoraussetzungen vorgesehenen Betriebssystemen sollten die aktuellen Service Packs installiert sein.

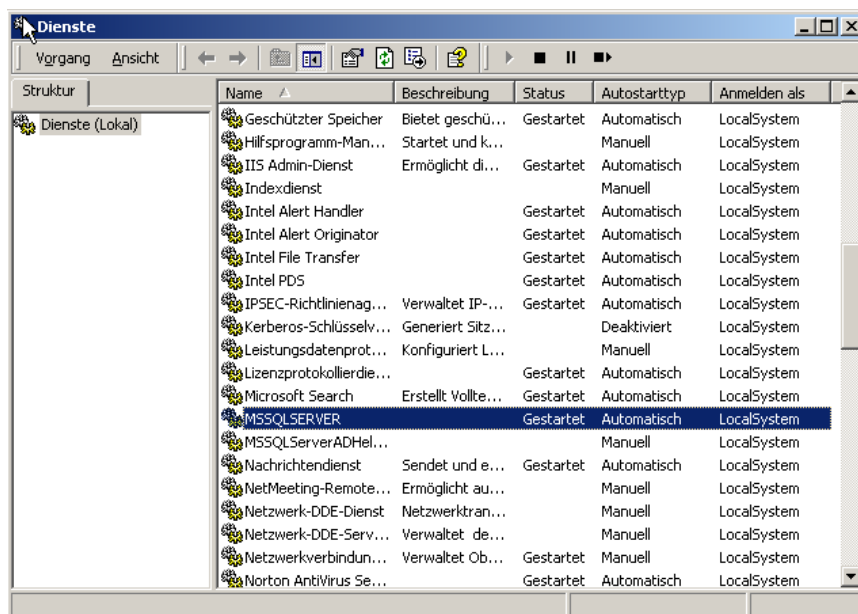


Abbildung 21: Dienste

Die Wahl des Dateisystems (NTFS oder FAT) hat keine Auswirkungen auf die Performance. NTFS ist aus Sicherheitsgründen jedoch FAT vorzuziehen.

Sämtliche nicht zwingend erforderlichen Dienste sind abzuschalten. Hierzu gehören insbesondere Warndienste, der Ablagemappen-Server, der Nachrichtendienst, Schedule und Virenschanner.

Ein paralleler Betrieb des Rechners mit dem SQL Server als Druck-, Web-, DHCP-, File- oder Applikations-Server ist nicht zu empfehlen. Insbesondere MS Exchange sollte auf einem separaten Rechner installiert werden. Die laufenden Dienste können u.a. in der Systemsteuerung überprüft werden.

5.1.3 Netzwerkkonfiguration

Es sollten nur die notwendigen Protokolle auf dem SQL Server installiert sein. Völlig ausreichend ist hier TCP/IP. Ferner sind auch die Netzwerkdienste auf ein absolut notwendiges Maß zu reduzieren. Novell-Dienste wie NetBIOS-Schnittstelle oder Novell Arbeitsstationsmanager sind - falls möglich - zu entfernen.

Die Eigenschaft des Server-Dienstes ist auf die Einstellung DURCHSATZ FÜR NETZWERKANWENDUNGEN OPTIMIEREN ZU setzen.

Der Rechner, auf dem der SQL Server läuft, ist nicht als PDC oder BDC geeignet. Der Netzwerkdienst DDE ist zu deaktivieren.

5.2 Optimierung

5.2.1 SQL Server - Allgemein

Der Speicherort der Windows-Auslagerungsdatei sollte sich auf einer anderen Festplatte als die MS SQL Server Programminstallation befinden.

Nicht zwingend erforderliche Dienste wie MSDTC, Microsoft Search oder English Query brauchen nicht installiert zu werden.

Die Ablage der Daten sollte separat von den Programmdateien auf einer weiteren Festplatte mit eigenem Festplattencontroller erfolgen. Es sollte sich hierbei jedoch nicht um die Festplatte handeln, auf dem sich die Windows-Auslagerungsdatei befindet.

5.2.2 Optimierung der SQL Server - Einstellungen

Die Dienste sind u. a. im **Dienst Manager** einzustellen und abzufragen. Ausschließlich die Dienste MSSQLServer (im SQL Server-Dienst-Manager als "SQL Server" angezeigt) und SQLServerAgent sind erforderlich. Die Dienste Distributed Transaction Coordinator und Microsoft Search sollten - falls installiert - deaktiviert werden. Dabei ist bei diesen Diensten die Option DIENST BEI BETRIEBSSYSTEMSTART AUTOMATISCH STARTEN natürlich ebenfalls zu deaktivieren.



Abbildung 22: SQL Server-Dienst-Manager (SQL Server 2000)

5.2.3 SQL Server - Eigenschaften

Die folgenden Optimierungen sind im **Enterprise Manager** vorzunehmen. Hierzu sind die Eigenschaften des registrierten SQL Servers der SQL Gruppe zu aktivieren.

Auf der Registerkarte ALLGEMEIN ist bei den Autostart Richtlinien für den Betriebssystemstart MSDTC zu deaktivieren.

Auf der Registerkarte SPEICHER kann die Option PHYSISCHEN SPEICHER FÜR SQL SERVER RESERVIEREN aktiviert werden.

Grundsätzlich ist die standardmäßig vorgeschlagene dynamische Konfiguration des MS SQL Server-Speichers beizubehalten. Hier wird dann das Maximum des Hauptspeichers auch im vollen Umfang für den SQL Server freigegeben. Dies steht in direktem Konflikt mit anderen Anwendungen aber auch mit dem Betriebssystem selbst. Daher ist das Maximum des Hauptspeichers (zweiter Schieberegler) zu beschränken, z.B. 50 MB weniger als der Gesamtspeicher zu wählen.

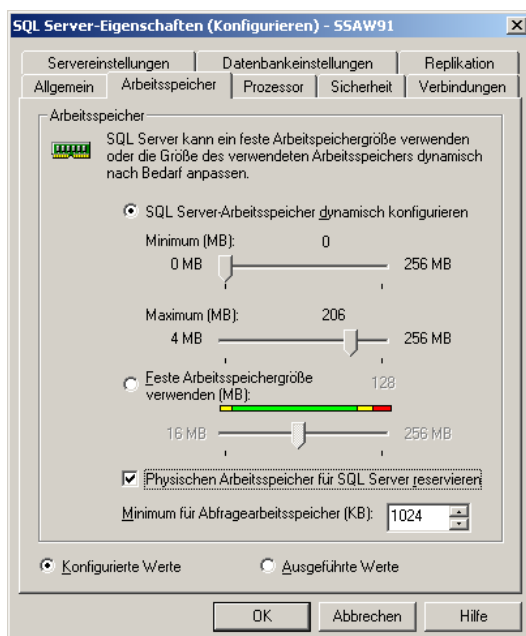


Abbildung 23: SQL Server-Eigenschaften (Konfigurieren) | Registerkarte Arbeitsspeicher

Die Option SQL SERVER PRIORITÄT AUF WINDOWS NT HÖHER STUFEN ist auf der Registerkarte PROZESSOR zu finden und zwingend zu aktivieren. Von der Einstellung WINDOWS NT-FIBER VERWENDEN raten wir hingegen ab.

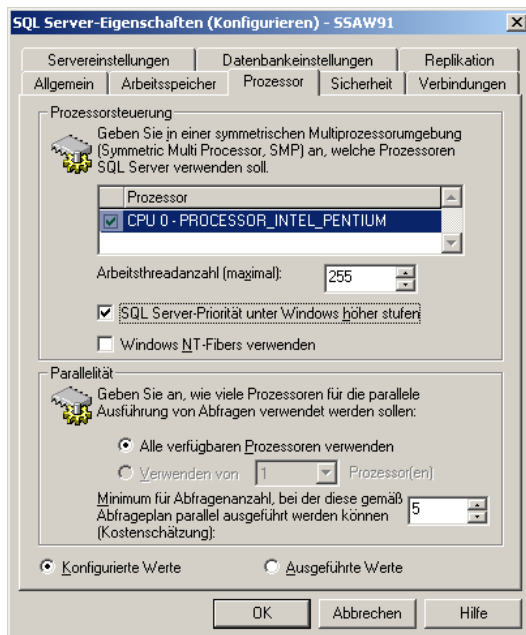


Abbildung 24: SQL Server-Eigenschaften (Konfigurieren) | Registerkarte Prozessor

5.2.4 Netzwerkoptimierung des Servers

Diese Konfigurationseinstellungen sind durch das Tool **SQL Server-Netzwerkconfiguration** (Server Network Utility) nach der Installation änderbar.

Es sind bereits bei der Installation nur die Protokolle TCP/IP und Named Pipes auszuwählen. Für den MS SQL Server 7.0 wurde zusätzlich noch Multiprotokoll empfohlen. Dies ist beim MS SQL Server 2000 nicht mehr erforderlich.

Die Protokollverschlüsselungen sind standardmäßig deaktiviert. Dies ist in Trusted Areas üblich. Die Konfigurationseinstellungen des MS SQL Servers 2000 sind auch in den Konfigurationseigenschaften des MS SQL Servers im Enterprise Manager durch Aktivierung der Schaltfläche NETZWERKKONFIGURATION der Registerkarte ALLGEMEIN aktivierbar.

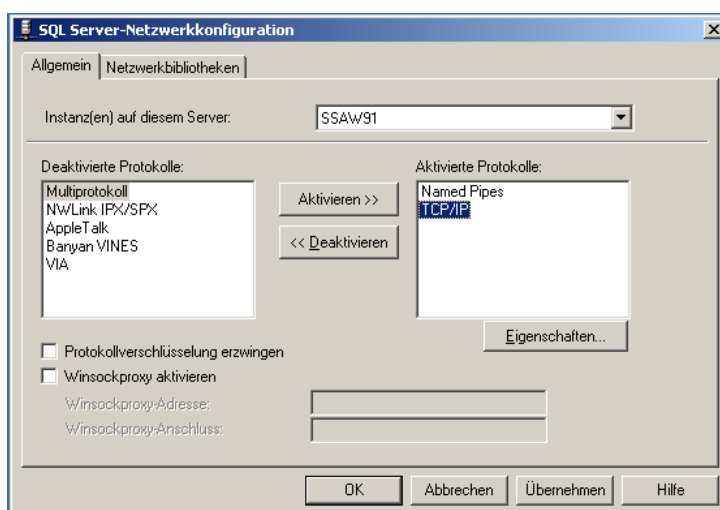


Abbildung 25: SQL Server-Netzwerkconfiguration | Allgemein

5.2.5 Netzwerkoptimierung Client

Diese Konfigurationseinstellungen sind durch das Tool **SQL Client-Netzwerkkonfiguration** (Client Network Utility) nach der Installation änderbar. Es gelten sämtliche Ausführungen für die SQL Server-Netzwerkkonfiguration.

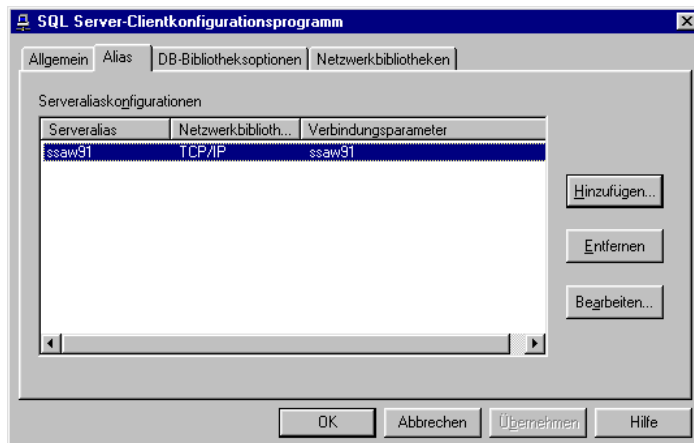


Abbildung 26: SQL Client-Netzwerkkonfiguration

Wir bemühen uns, diese Dokumentation ständig zu verbessern. Sollten Sie Verbesserungsvorschläge, Anregungen oder möglicherweise Fehlerhinweise an uns weiterleiten wollen, so schicken Sie uns bitte einfach eine Mail an technik@wac.de mit dem Betreff DATENSICHERUNG UND EINSATZ SQL SERVER.

Viel Erfolg mit tse:nit

Ihr tse:nit - Team

(c)2011 ADDISON Software und Service GmbH