

Datenbank über einen einheitlichen DNS-Namen / IP-Adresse ansprechen. Die Applikation, die die Datenbank nutzt, hat somit Transparenz über einen eventuell erfolgenden Failover und muss nicht zur Laufzeit einen anderen Datenbankserver ansprechen.

Cloud Backup

Das Microsoft® SQL Server® Backup to Microsoft Azure® Tool erlaubt eine verschlüsselte und komprimierte Sicherung in die Microsoft Cloud. Möglich ab Windows 7 SP1 und Windows Server 2008 R2 SP1. Voraussetzung dafür ist der Microsoft Cloud Zugang, setzt man auf die Backup Strategie, ist man mit den Bestimmungen einverstanden und die aktuellen Kosten (*Azure pricing calculator for database management* verwenden) stehen im Verhältnis zum Nutzen.

SQL Server Maintainance Plan

Natürlich verfügt der SQL Server über einen Wizzard gesteuerten Wartungsplan, bei dem am Ende aller getroffenen Einstellungen ein Script vom Agent ausgeführt wird. Bei wem funktioniert der DB CleanUp Task?

Für den professionellen Einsatz würde ich dann eher die mehrfach ausgezeichneten *SQL Server SQL Stored Procedures* von **Ola Hallengren** [1], den schwedischen SQL DBA, empfehlen. Sie bietet mehr Optionen an und geht sparsamer mit den Ressourcen um.

Disaster Recovery

RPO - Recovery Point Objective

Wie viel Datenverlust kann verkraftet werden.

Es handelt sich um den Zeitraum, der zwischen zwei Datensicherungen liegen darf. Also wie viele Datentransaktionen dürfen zwischen der letzten Sicherung und des möglichen Systemausfalles verloren gehen. Sollen keine Daten verloren gehen, liegt der Wert für RPO bei 0 Sekunden.

RTO - Recovery Time Objective

Wie lange darf ein Geschäftsprozess ausfallen.

Es handelt sich um jene Zeit von Eintritt des Schadens bis zur vollständigen Wiederherstellung des Geschäftsprozesses. Der Zeitpunkt kann von 0 Minuten bis Tage oder Wochen dauern.

Auswahl seines passenden Backup Plans

RPO > 0

Hier kann *Backup und Restore* zum Einsatz kommen, das ist auch schon mit der SQL Express Version möglich. Es funktioniert auch im *Recovery Modus Simple*. RPO und RTO sind dabei von der Datenmenge abhängig. Auch der Einsatz eines Hot Standby Systems ist möglich.

Setzt man *Log Shipping* ein, hat dies den Vorteil einer kurzen RTO, man ist aber von der Datenmenge abhängig. Man hat weniger Einfluss auf das Primärsystem, und es können mehrere Hot Standby Systeme eingesetzt werden. Allerdings benötigt

man dazu min. die SQL Server Standard Edition und hier muss der Recovery Modus auf Full eingestellt sein. Es ist ein manueller Eingriff für den DB Admin notwendig, der ähnlich wie bei Backup und Restore vor sich geht. Ein RPO von wenigen Minuten kann damit erreicht werden.

Beim Einsatz von *Replication* ist der RTO defakto null. Am SQL Server funktioniert es auch im *Recovery Model Simple* ab der *Standard Edition*. Es sind mehrere Subscriber möglich, es ist auch als Scale Out System verwendbar. Aber der RPO ist unberechenbar, vor allem bei hoher Last. Es kann einen hohen Einfluss auf das Primärsystem haben, aber vor allem auf das Transaction Log File.

RPO = 0

SQL Failover Cluster kommt auf einen RTO von wenigen Sekunden. Es ist die einzige Lösung, bei der auch die Server Objekte, also *Linked Server* und die *Agent Jobs*, weiterhin funktionieren. Diese Lösung ist transparent für die Applikationen. Der SQL Server unterstützt dies aber der Standard Edition. Diese komplexe Lösung benötigt aber viel Infrastruktur wie *Active Directory* und *Shares Storage* und die Windows und SQL Server Versionen muss auf allen Nodes die gleiche sein.

Database mirroring kommt auf einen RTO von wenigen Sekunden. Es wird keine aufwändige Infrastruktur benötigt und ist aber der SQL Server Standard Edition möglich. Diese Technik ist in der MSDN bereits als deprecated beschrieben und wird in Zukunft durch AO ersetzt werden. Die Nachteile sind außerdem, dass ein spezieller Treiber am Client benötigt wird. Die Geschwindigkeit des Mirrors kann einen Einfluss auf den Primary haben. Am SQL Server ist Recovery Modus Full zu verwenden.

Availability Groups haben einen RTO von wenigen Sekunden. Es wird eine spezielle Infrastruktur vorausgesetzt, für den Client wirkt es transparent. Funktioniert leider nur in der SQL Server Enterprise Edition und den *Recovery Modus Full*. Es handelt sich um eine komplexe Lösung aufgrund mehrerer Komponenten. Auch hier müssen Windows und SQL Server Versionen auf allen Nodes die gleiche sein. Die Geschwindigkeit der Replica hat zudem Einfluss auf den Primary Server.

Virtualisierungcluster

Mittels eines Virtualisierungclusters kommt man auf einen RTO von null bei einem geplanten Failover und einem RPO von null. Es ist transparent für SQL Server und für den Client. Diese Lösung funktioniert für alle SQL Server Edition, auch bei der Express Version.

Nachteil ist hier eine längere Recovery Zeit bei einem ungeplanten Failover. Bei dieser komplexen Lösung ist viel Infrastruktur notwendig.

Virtualisierungsreplica

Mittels eines Virtualisierungsreplica kommt man auf einen RTO und RPO von null bei einem geplanten Failover und es ist ebenfalls transparent für SQL Server und für den Client. Allerdings kann es zu einem Datenverlust bei einem ungeplanten Failover kommen. Auch diese Lösung ist komplex durch die hohe Anzahl von Komponenten.

Durchspielen seines gewählten Szenarios

Der Beste theoretische Plan hilft nichts, wenn er nicht zumindest einmal durchgezogen wurde. Es gibt Fälle wo das Backup defekt oder nicht mehr kompatibel zum aktuellen Environment ist. Aber auch ganz triviale, wo es eine oder keine Urlaubsvertretung gibt und der Backup Job nicht läuft oder die passenden Medien nicht rechtzeitig gefunden oder darauf zugegriffen werden können.

Links & Quellen

[1] <https://ola.hallengren.com/>

Fazit

Es gibt unzählige Varianten der Datensicherung, wobei die passende Lösung von der Anwendung und von der Infrastruktur abhängig ist. Eine Hochverfügbarkeit führt zu einem höheren Wartungsaufwand und setzt auch Wissen über mehrere Themenbereiche voraus. Wird die kostenintensive Hochverfügbarkeit schlecht gewartet, führt dies am Ende zu einem noch größeren Problem.

Autorenbox

Thomas Reinwart verfügt über umfangreiche Berufserfahrung auf dem IT Sektor. In den letzten 20 Jahren war er in den Bereichen Softwareentwicklung, Software-design, Architekt und als Consultant tätig. Technischer Fokus ist derzeit Microsoft .net und SQL Server, wo er alle aktuellen Microsoft Zertifizierungen hat.

E-Mail: office@reinwart.com

