

Kapitel 6

Ganz SQL ist eine Wolke – SQL Azure und andere Cloud-Optionen

In diesem Kapitel:

| | |
|---|-----|
| Überblick über Windows Azure | 202 |
| Windows Azure Datenbank – Die Datenbank in der Wolke | 207 |
| Office 365: Business Intelligence mit SharePoint Online | 221 |
| Zusammenfassung | 221 |

Die Datenbanken von StarBack sind nun schon einige Zeit auf Microsoft SQL Server in Betrieb, es werden weiterhin Applikationen migriert und die Last auf dem Server sowie der Speicherplatzverbrauch erhöhen sich. Die Administratoren haben aber alles im Griff, denn dank des Einsatzes der modernen System Center Suite können Voraussagen über die Last und das Wachstum der Datenbanken getroffen werden. Frühzeitig wird der Server aufgerüstet oder Plattenplatz im SAN nachgerüstet.

Der IT-Leiter fragt sich allerdings, ob man die Kosten im Betrieb nicht auch senken könnte und blickt etwas über den Tellerrand des einzelnen Servers hinaus. Und da findet sich noch eine weitere Komponente. In der letzten Zeit wurden die Vertriebsmitarbeiter mit Windows 8 Tablet-Computern ausgestattet, um so per Excel von überall aus über ihre Mobilfunkverbindung auf die Daten der jeweiligen Kunden zugreifen zu können. So können in Gesprächen aktuelle Lieferkonditionen sowie die Historie eingesehen werden, was hilfreich ist, wenn der Kunde über Probleme sprechen möchte und so eine ganz bestimmte Lieferung gleich auf dem Bildschirm auftauchen kann.

So nützlich dies auch ist, stieg der externe Datenverkehr von StarBack massiv an und da der Hauptsitz in einem Außenbezirk liegt, sind auch die Preise für die Datenleitung nicht gerade gering. Da lassen sich gleich zwei Fliegen mit einer Klappe schlagen, wenn genau diese Daten in die Cloud verlagert werden. Dadurch lässt sich Speicherplatz für die Data Marts sparen und die Daten müssen nur einmal in die Cloud synchronisiert werden. Die Antwort des SQL Servers lautet darauf: Windows Azure Datenbank!

Überblick über Windows Azure

Seit ein paar Jahren ist dieses Buzzwort in aller Munde: Man spricht von der Datenwolke, oft auch schlicht die *Cloud* genannt. Was für den einen ein undurchsichtiges Kondensat von Wassermolekülen ist, ist für andere der heilige Gral zu Hochverfügbarkeit und Skalierbarkeit bei vertretbaren Kosten. Auf diesen Zug ist auch Microsoft aufgesprungen und hat am 1. Februar 2010 erstmals Kapazitäten unter dem Namen *Windows Azure* angeboten.

Windows Azure ist das Angebot von Microsoft, Ressourcen als *Platform as a Service (PaaS)* in der Wolke anzumieten. PaaS bedeutet, dass der Anbieter sich um die Bereitstellung der Anwendungsinfrastruktur, sprich die Hardware, Betriebssystem, Netzwerkanbindung und Softwareframeworks kümmert, sodass Sie sich vollständig auf die Entwicklung Ihrer Anwendung konzentrieren können. Die Ressourcen werden in Europa mittels Rechenzentren sowohl in Irland als auch in Amsterdam bereitgestellt und bieten über eine Synchronisation zwischen den Rechenzentren auch eine geografische Ausfallsicherheit. Genauer gesagt bietet Microsoft mit Azure direkt an, Rechenzeit in Form von CPU-Zeit und Arbeitsspeichergrößen anzumieten, Webdienste zu hosten, virtuelle Hyper-V-Images bereitzustellen und sogar SQL-Datenbanken!

Sie mögen sich vielleicht fragen, wie es mit Office 365 und weiteren Onlinediensten wie Windows Intune aussieht. Microsoft trennt in dem Angebot zwischen angemieteten *Ressourcen* und angemieteten *Diensten*. Verschaffen wir uns erst einen Überblick über das gesamte Angebot, bevor wir uns tiefgehend mit SQL Azure beschäftigen.

HINWEIS

Microsoft hat die Benennung sowie Strukturierung des Windows Azure-Angebots mehrfach komplett überarbeitet. Wir beziehen uns in diesem Kapitel auf den Stand vom Ende Oktober 2012. Wir verweisen aber auch hin und wieder auf bereits jetzt schon veraltete Begrifflichkeiten, wobei diese vor 14 Tagen noch gültig waren. Man muss sich aktuell auf teilweise monatliche Erneuerungszyklen einstellen.

Mieten Sie einen CPU-Kern

In Windows Azure können Sie Rechenzeit anmieten, indem Sie sich für *Cloud-Dienste* entscheiden (ehemals Windows Azure Compute). Microsoft bietet Ihnen dort die Möglichkeit, Anwendungen und Dienste mit dem Windows Azure SDK zu entwickeln und zu veröffentlichen.

Ein Cloud-Dienst kann entweder im Kontext von den Internetinformationsdiensten (IIS) oder auch direkt unter Windows als so genannte Worker Role (Workerrolle) bereitgestellt werden. Was vor ein paar Jahren kaum vorstellbar war, ist die Breite des Angebots. Microsoft beschränkt sich bei dem Angebot der Frameworks nicht nur auf .NET, Sie können auch Oracle's Java und sogar die quelloffene Sprache PHP verwenden.

Wenn Sie Ihre Anwendung mit dem Windows Azure SDK für den Betrieb in der Wolke vorbereitet und veröffentlicht haben, können Sie sich entscheiden, wie viel Rechenleistung Sie verwenden möchten. Dabei stehen Ihnen fünf verschiedene Skalierungsstufen zur Verfügung.

Die Bereitstellung der Ressourcen ist durch das Hochfahren einer dedizierten virtuellen Maschine (VM) für Ihre Cloud-Dienste gelöst worden. Das heißt, Sie teilen sich die Umgebung nicht mit anderen Kunden und können somit auch nicht von Konfigurationsfehlern oder Betriebsstörungen anderer Anwendungen getroffen werden.

| Größe der VM | CPU-Kern | RAM in GB | Lokaler Speicher in MB | Bandbreite in MBit/s |
|--------------|----------|-----------|------------------------|----------------------|
| Extra Klein | Geteilt | 768 (MB) | 19.480 | 5 |
| Klein | 1 | 1,75 | 229.400 | 100 |
| Mittel | 2 | 3,50 | 500.760 | 200 |
| Groß | 3 | 7,00 | 1.023.000 | 400 |
| Extra Groß | 4 | 14,00 | 2.087.960 | 800 |

Tabelle 6.1 Angebotene Instanzgrößen für Windows Azure Cloud-Dienste

Sie bekommen, außer bei der extra kleinen Instanz, sogar dedizierte Prozessorkerne zugeteilt, die Sie mit keiner anderen virtuellen Umgebung teilen. Auch wenn Sie eine eigene virtuelle Maschine erhalten, haben Sie im Rahmen der Cloud-Dienste keinen direkten Zugriff auf das Betriebssystem. Microsoft kümmert sich um das Einspielen von Updates und die Pflege des Betriebssystems selbstständig.

ACHTUNG Das heißt aber auch, dass die Installation von zusätzlichen Softwareprodukten nicht möglich ist. Zum Registrieren von Assemblys im Globalen Assemblycache (GAC) müssen Sie den Microsoft-Support kontaktieren.

Abgerechnet wird die Nutzung der Cloud-Dienste pro bereitgestellter Stunde und dann in das Leistungsäquivalent der kleinen Instanz umgerechnet. Besonders interessant ist dieses Modell für Anwendungen, die nicht permanent betrieben werden müssen oder nur zu ganz bestimmten Tagen einer hohen Last ausgesetzt sind.

Hyper-V in der Wolke

Alternativ können Sie auch einen selbst verwalteten virtuellen Computer anmieten, dazu stehen Ihnen in Windows Azure die gleichen Instanzgrößen von extra klein bis extra groß zur Verfügung. Bei der Erstellung fragt Sie das Azure Portal auch nach dem Betriebssystem. Hier hat Microsoft die nächste Überraschung in petto, denn neben Windows Server 2008 R2, Windows Server 2012 und gebündelten Images mit BizTalk Server oder SQL Server begegnen einem auch CentOS, openSUSE, SUSE Linux Enterprise Server und Ubuntu Server.

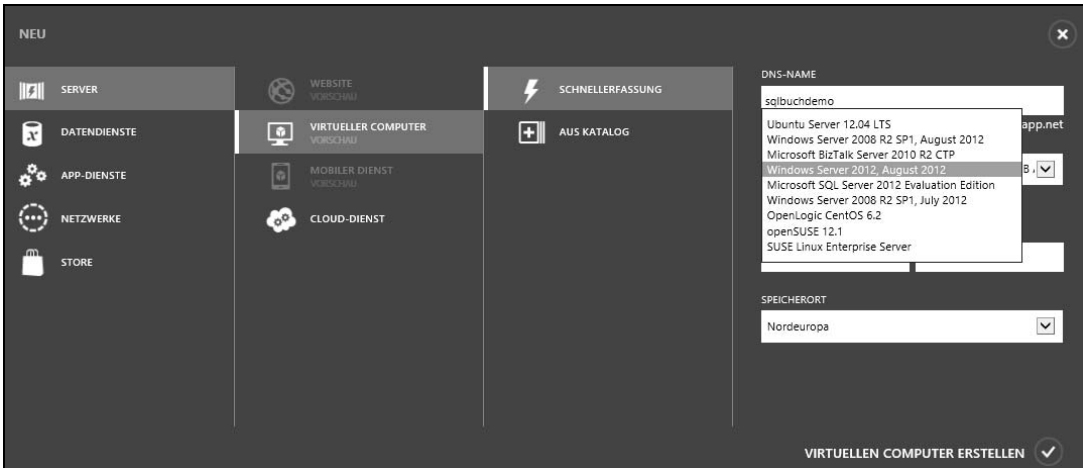


Abbildung 6.1 Auswählen von Betriebssystemen bei der Erstellung eines neuen virtuellen Computers

Nachdem Microsoft mit Windows Server 2008 R2 die Unterstützung von Linux-Gastsystemen in Hyper-V eingebaut hatte, wird mit Windows Azure nun einen Schritt weitergegangen und der Betrieb von Linux-basierenden virtuellen Computern im Microsoft-Rechenzentrum angeboten.

Neben der Freiheit, beliebige Komponenten auf dem System installieren zu können, bietet Microsoft für den BizTalk Server und den SQL Server auch das Abrechnungsmodell *pro Stunde* an. Sie müssen also nicht mehr unbedingt eine Lizenz erwerben und können nach Bedarf die Maschine und so auch die gemietete Lizenz nutzen. Im Gegensatz zu den Cloud-Diensten, müssen Sie sich hier allerdings um die Wartung des Systems selbst kümmern.

HINWEIS

Dieser Dienst befindet sich derzeit noch in der Vorschau. Microsoft behält sich vor, Dienste, die sich noch in der Vorschau befinden, umfangreich anzupassen oder auch ganz einzustellen.

Windows Azure als Datenablage für Big Data

Nach der Cloud ist eines der neueren Buzzworte *Big Data*. Ein Begriff, der zum gegenwärtigen Zeitpunkt mindestens genauso schwammig und vielseitig interpretierbar ist wie vor einigen Jahren, als die Cloud im Aufkommen war. Im Allgemeinen wird unter *Big Data* das Speichern, Verwalten und Analysieren von sehr großen unstrukturierten Datenmengen verstanden. Das Augenmerk liegt hier auf unstrukturierten Daten, ganz im Gegensatz zu den strukturierten Daten in einem Data Warehouse. Es kommt jedoch vereinzelt vor, dass auch sehr große Data Warehouses unter dem Begriff *Big Data* genannt werden.

Windows Azure bietet gegenwärtig Speicher an, der auch unter dem Namen *BLOB Storage* bekannt ist. Sie können im Azure-Speicher Schlüssel/Wert-Paare von bis zu mehreren hundert GB Größe erstellen und insgesamt bis zu 100 TB pro Konto speichern. Der Speicher eignet sich besonders für schwer strukturierbare Daten wie Bilder, Dokumente, Binärdateien, Audio und Video (Streaming) oder auch für Backups. Darüber hinaus legen andere Windows Azure-Dienste ihre Daten im Speicher ab, zum Beispiel virtuelle Festplatten der virtuellen Computer.

Daten im Speicher sind in Containern organisiert, so können Sie Ihre verschiedenen Daten im *Speicher* wie im folgenden Beispiel gruppieren.

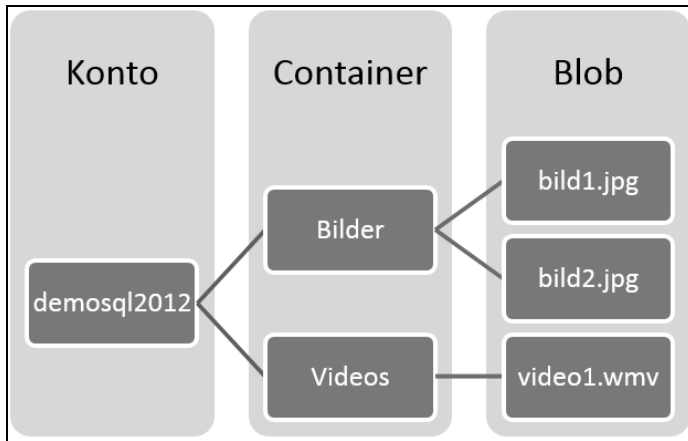


Abbildung 6.2 Hierarchie von Daten im Windows Azure-Speicher

Um Ihre Daten im Speicher zu betrachten, können Sie sich mit dem SQL Server Management Studio 2012 (SSMS) mit Windows Azure verbinden. Das Management Studio zeigt Ihnen allerdings nur die Strukturen und Dateilisten in den jeweiligen Containern.

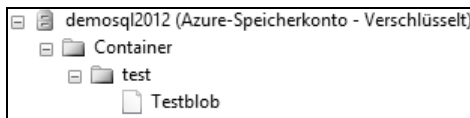


Abbildung 6.3 Zugriff auf den Speicher in SSMS

Sie können jedoch nicht die Daten herunterladen, manipulieren oder betrachten. Eine wesentlich benutzerfreundlichere Möglichkeit bietet das Werkzeug *Azure Storage Explorer* (<http://azurestorageexplorer.codeplex.com/>). Mit dem Azure Storage Explorer können Sie Container anlegen und ihre Daten bearbeiten oder löschen. In der folgenden Abbildung betrachten wir gerade den obigen Testblob, der wiederum ein Screenshot aus dem Kapitel 14 zu Data Quality Services ist.



Abbildung 6.4 Der Blob im Azure Storage Explorer

Eine weitere Möglichkeit, unstrukturierte Daten abzulegen, bietet die Software *Apache Hadoop*. Hadoop enthält einen von Google entwickelten MapReduce-Algorithmus und wird unter anderem von Facebook, AOL und Yahoo zur Datenspeicherung genutzt.

Derzeit integriert Microsoft Hadoop in Windows Azure und ist bereits als Vorschauprogramm verfügbar. Auch hier findet eine weitere OpenSource-Technologie Einzug in die Microsoft-Rechenzentren. Wir dürfen gespannt sein, welche neuen Möglichkeiten Windows Azure in Zukunft noch bieten wird.

Die neue Welle: Apps und mobile Dienste

Mit der Veröffentlichung von Windows 8 und Windows Phone 8 rückt auch die App-Programmierung, also die Implementierung von kleinen Anwendungen, die über mehrere Plattformen genutzt werden können, in den Fokus. Durch das einheitliche Bedienkonzept zwischen Windows Phone 7/8 und Windows 8 öffnen sich neue Möglichkeiten zur Verknüpfung von Anwendungen zwischen dem PC und Mobilgeräten.

Deshalb gibt es, ebenfalls noch in der Vorschau, Windows Azure Mobile Services. Mobile Services unterstützen die Entwickler von Apps, Zustände der Anwendung, wie Einstellungen oder auch HighScore-Listen, nicht nur lokal zu speichern. So können die Zustände zwischen den verschiedenen Plattformen einheitlich genutzt werden und die Nutzer bei Änderungen auch über Push benachrichtigt werden. Die Integration in die eigene App erfordert dabei nur wenige Zeilen Code.

Wie sicher ist das eigentlich?

Der Gedanke, neue Anwendungen auf Basis der Cloud aufzubauen, wirkt aufgrund der Einfachheit schnell charmant. Oftmals erzeugt der Gedanke an die Datensicherheit aber ein schlechtes Bauchgefühl. Der Transfer von Daten über das Internet stellt schnell ein Sicherheitsrisiko dar. Daten können abgehört werden oder auch durch *Man-in-the-Middle*-Attacken manipuliert werden.

Der Zugriff über HTTPS erschwert diese Art der Manipulation schon erheblich, dennoch werden die Daten weiterhin neben dem weiteren Internetverkehr parallel über dieselben Routen geleitet. Hier hat Microsoft nachgebessert und eine Netzwerkverwaltung aufgebaut. Sie können nun sowohl innerhalb Ihrer gemieteten Dienste in Windows Azure virtuelle Netzwerke aufbauen, als auch Ihr Netzwerk oder das Ihrer Kunden an ein solches virtuelles Netzwerk über eine IPsec Verbindung anbinden. Dieser Dienst nennt sich *Windows Azure Connect*.

Windows Azure Datenbank – Die Datenbank in der Wolke

Was wäre eine Cloud ohne eine relationale Datenbank? Viele Anwendungen, seien es Webdienste oder Anwendungen, die unter *Cloud-Dienste* fallen, benötigen eine relationale Datenbank zur Speicherung ihrer Datensätze. Weitere Szenarien, wie verteilte Anwendungen, die über das Internet auf eine Datenbank zugreifen, oder auch einfach als Sicherung von Daten einer lokalen Datenbank, sind zusammen mit einer relationalen Datenbank in der Wolke denkbar.

Für Microsoft ist es natürlich naheliegend, eine Version von SQL Server in Windows Azure anzubieten. Dieser Dienst, bisher unter dem Namen *SQLAzure* bekannt, wird nun als die *Windows Azure SQL-Datenbank* vermarktet. Es handelt sich jedoch nicht einfach um eine in der Cloud installierte Version von SQL Server, sondern um eine ganz speziell angepasste Variante.

Jede Windows Azure SQL-Datenbank wird dreifach in einem Rechenzentrum vorgehalten und permanent synchronisiert. So ist ihre Datenbank automatisch hochverfügbar und bleibt bei dem Ausfall einer Instanz weiterhin erreichbar. Gleichzeitig kümmert sich Windows Azure um die optimale Ablage Ihrer Datenbankdateien und Transaktionsprotokolle, damit es zu möglichst wenigen Wartezeiten bei der Transaktionsverarbeitung kommt.

Wird eine Verbindung zu Ihrer Windows Azure SQL-Datenbank aufgebaut, kümmert sich ein Tabular Data Stream (TDS)-Gateway über die Zuteilung Ihrer Verbindung zu einer laufenden Azure-Datenbankinstanz. TDS ist übrigens das Protokoll, mit dem generell auf den SQL Server und nicht nur auf Windows Azure SQL-Datenbanken zugegriffen wird.

Um diese Architektur implementieren zu können, hat die SQL-Datenbank nicht dieselbe Codebasis von SQL Server. Die meisten Funktionalitäten werden zwar angeboten und vielfach kann vorhandener Transact-SQL-Code nach einer Migration zu Windows Azure SQL-Datenbank wiederverwendet werden, es gibt aber auch eine ganze Reihe an Features und Transact-SQL-Ausdrücken, die bisher nicht angeboten wird. Microsoft entwickelt die Windows Azure SQL-Datenbank stetig weiter und rüstet immer mehr Funktionalitäten von SQL Server nach.

Einschränkungen der Windows Azure SQL-Datenbank

Wenn Sie auf der Windows Azure SQL-Datenbank arbeiten, werden Sie feststellen, dass sämtliche Verwaltungsfunktionen wie administrative Views, administrative gespeicherte Prozeduren, Ressourcenkontrolle (Resource Governor), Auditing oder das Erstellen von Dateigruppen nicht möglich ist. Das ist aber gut und richtig so, da Ihnen diese Arbeit von Windows Azure schon abgenommen wird und sich das Verhalten der Datenbank der Cloud gegenüber einer lokal installierten Lösung unterscheidet.

HINWEIS

Gegenwärtig ist das gesamte Thema .NET Framework in der Windows Azure SQL-Datenbank noch nicht verfügbar. Das heißt, dass die Geodatentypen sowie Hierarchy nicht angeboten werden, ebenso ist es nicht möglich, eigene .NET-Assemblys bereitzustellen. Sie müssen sich daher vorerst ausschließlich mit Transact-SQL bewaffnen.

Die beiden bisher genannten Einschränkungen müssen Sie nicht unbedingt treffen, eine Sache die auf nahezu jede Datenbank zutrifft, ist das Vorkommen von Heaptabellen, d.h. Tabellen, die über keinen gruppierten Index verfügen. Wenn Sie eine Tabelle in die Windows Azure SQL-Datenbank anlegen oder migrieren, muss immer ein gruppierter Index angelegt werden. Beachten Sie dies bei Ihrem Datenbankdesign.

Es gibt eine Reihe von Transact-SQL Befehlen, die bisher nicht angeboten werden. Diese sind überwiegend die Folge aus den oben beschriebenen Einschränkungen. Ein paar weitere fehlende Funktionen sind globale temporäre Tabellen und Trace Flags (Ablaufverfolgungsflags). Eine komplette Liste gibt es in der MSDN unter <http://msdn.microsoft.com/en-us/library/ee336253.aspx>.

Eine weitere, sehr wesentliche Einschränkung, ist die gegenwärtige Beschränkung der maximalen Datenbankgröße auf 150 GB. Wenn Ihnen diese nicht ausreicht, können Sie aber mit *Database Federations* Ihre Datenbank partitionieren und die Partitionen über mehrere Datenbanken aufteilen. Dann gilt die Beschränkung von 150 GB Daten pro Partition. Mehr dazu weiter hinten in diesem Kapitel.

Die Datenbank wird eingerichtet

Im Windows Azure Management Portal können Sie eine Datenbank genauso wie die zuvor beschriebenen Dienste anlegen. Das Portal fragt Sie bei dem Anlegen einer Datenbank nach der Edition, der gewünschten Sortierreihenfolge sowie in welcher Region Sie einen Server für diese Datenbank betreiben möchten.



Abbildung 6.5 Erstellen einer Datenbank in Windows Azure

Nach dem Betätigen des Häkchens in der unteren, rechten Seite dauert es ein paar Minuten und Ihre neue Windows Azure SQL-Datenbank steht Ihnen zur Verfügung.

Die Verbindung wird hergestellt

Bevor Sie eine Verbindung zu Ihrer Windows Azure SQL-Datenbank herstellen können, müssen Sie im Management Portal Ihre IP-Adresse für den Zugriff auf die Datenbank freischalten. Grundsätzlich verhindert die Firewall einen Zugriff auf die Datenbank, Sie können nicht nur einzelne IP-Adressen, sondern auch Adressbereiche, beispielsweise eines über Windows Azure Connect angebundenes Netzwerkes, freischalten.

Den Administratorbereich Ihrer Datenbank erreichen Sie im Management Portal, indem Sie auf die Registerkarte *Konfigurieren* klicken. Alternativ ermöglicht es auch die Prozedur `sp_set_firewall_rule`, die Firewallregeln zu überarbeiten.

```
USE [master]
EXEC ,Zu hause', ,111.222.333.444', '111.222.333.444'
```

Listing 6.1 Hinzufügen eines Eintrags zur Firewallkonfiguration der Windows Azure SQL-Datenbank

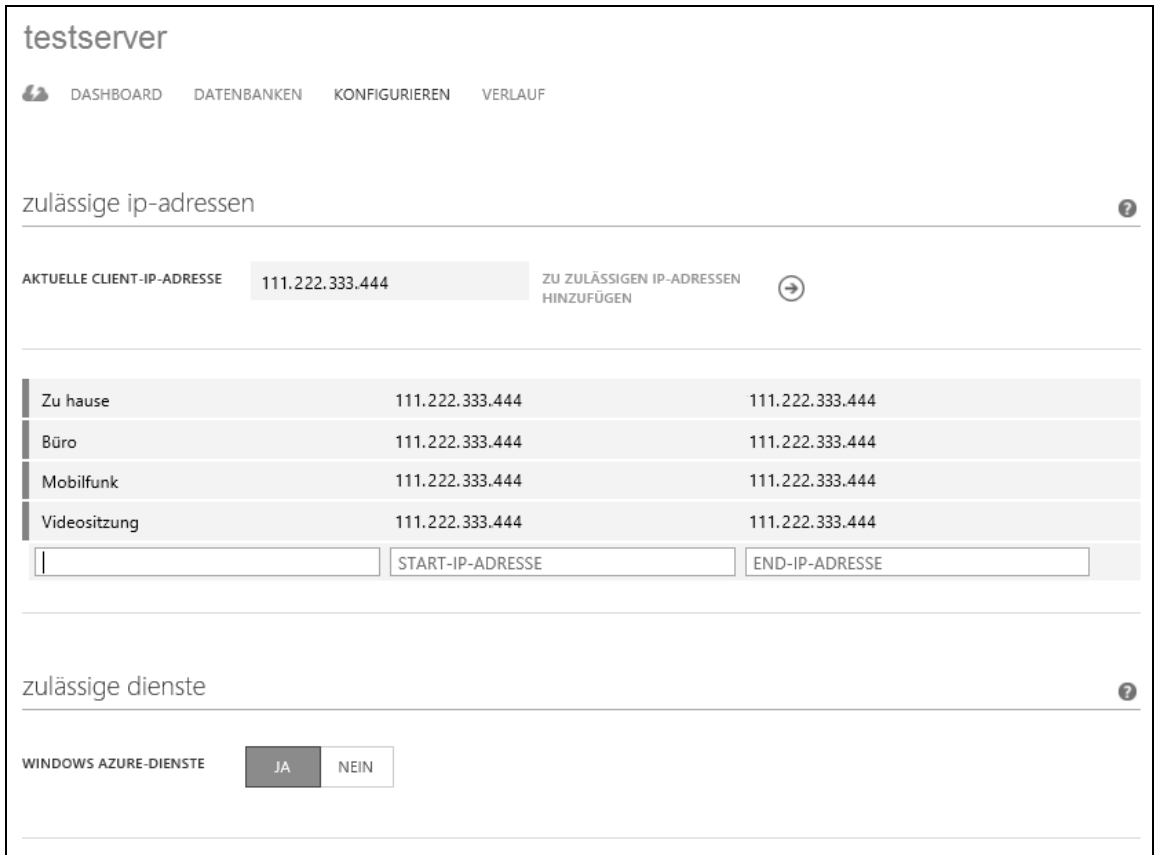


Abbildung 6.6 Firewallregeln der SQL-Datenbank

Nun können sie sich mit dem Management Studio wie mit einer gewöhnlichen Datenbankinstanz verbinden. Sobald die Verbindung aufgebaut ist, zeigen sich aber auch bereits die ersten Unterschiede.

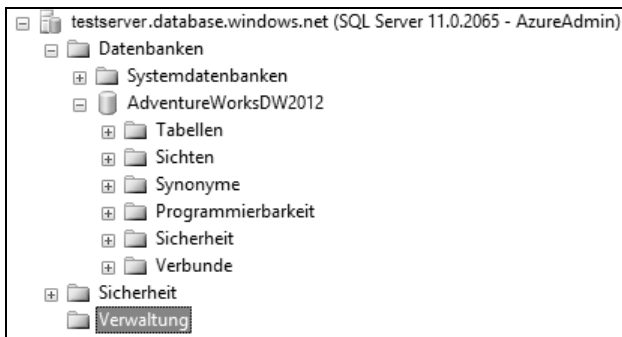


Abbildung 6.7 Verbindung mit einer Azure SQL-Datenbank im Management Studio

Wie bereits beschrieben, sind die administrativen Funktionen im Punkt *Verwaltung* aus gutem Grund nicht verfügbar. Eine Verbindung zur Windows Azure SQL-Datenbank lässt sich auf den ersten Blick nicht nur daran erkennen, sondern auch an dem grauen Serversymbol mit einer blauen Datenbank, während eine Verbindung zu einer normalen Datenbankinstanz von SQL Server mit einem gelben Datenbanksymbol gekennzeichnet wird. Ansonsten stehen Ihnen die vorhandenen Funktionalitäten im Management Studio wie von einer lokalen oder in Ihrem Rechenzentrum installierten Instanz gewohnt zur Verfügung.

Sie können eine Windows Azure SQL-Datenbank aber auch ohne Verwendung von Management Studio verwalten, erstellen und bearbeiten. Um zu dem Verwaltungsbereich im Windows Azure-Managementportal zu gelangen, markieren Sie Ihre SQL-Datenbank und klicken auf *Manage*. Sie müssen sich dann mit dem administrativen Benutzer an Ihrer Azure-Datenbank anmelden. Nach dem Einloggen zeigt das Portal eine Übersicht zum Status Ihrer Datenbank.

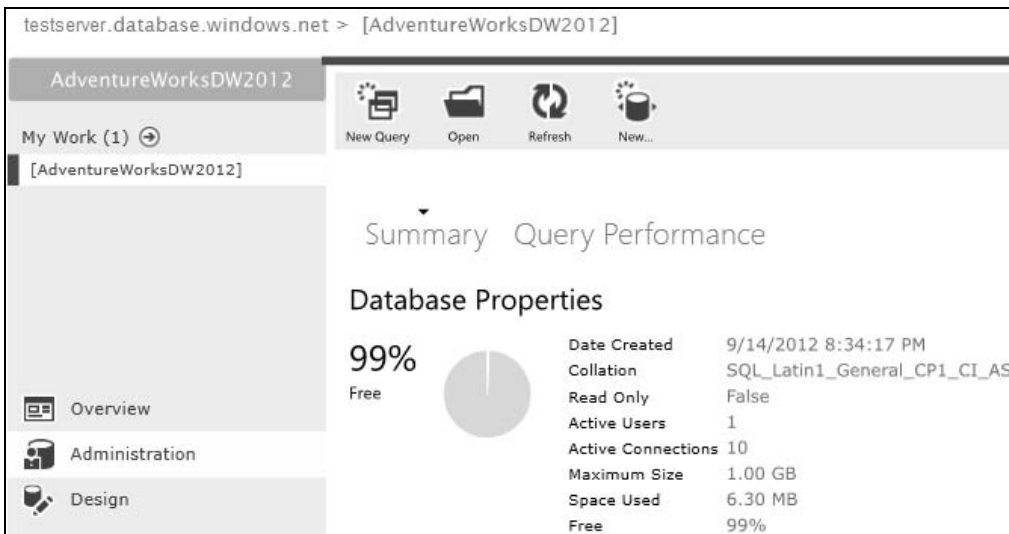


Abbildung 6.8 Datenbankübersicht im Windows Azure Management Portal

Auf einen Blick können Sie sehen, wie viele Benutzer und Verbindungen aktiv sind und wie viel Speicher Sie noch frei haben. Probieren Sie auch den Menüpunkt *Query Performance*. Dort sehen Sie die letzten Abfragen auf Ihre Datenbank inklusive der Werte für CPU-Zeit, physische und logische Lese-/Schreibvorgänge. Das alles ohne Abfragen auf Management Views bauen zu müssen, ist wirklich sehr praktisch.

Wenn Sie in der linken Leiste auf den Menüpunkt *Design* wechseln, erhalten Sie eine Übersicht Ihrer Datenbanktabellen und können diese bearbeiten und natürlich auch neue erstellen oder vorhandene löschen. Das sind soweit gewöhnliche Funktionalitäten, die genauso wie im Management Studio funktionieren. Etwas Besonderes bietet Ihnen die Oberfläche an dieser Stelle aber auch: Fahren Sie mit der Maus über eine Ihrer Tabellen, sehen Sie rechts nicht nur eine Schaltfläche für *Edit* sondern auch *Dependencies*.

Im folgenden Beispiel haben wir einen Teil der *AdventureWorks2012*-Schemata in eine Windows Azure SQL-Datenbank importiert.

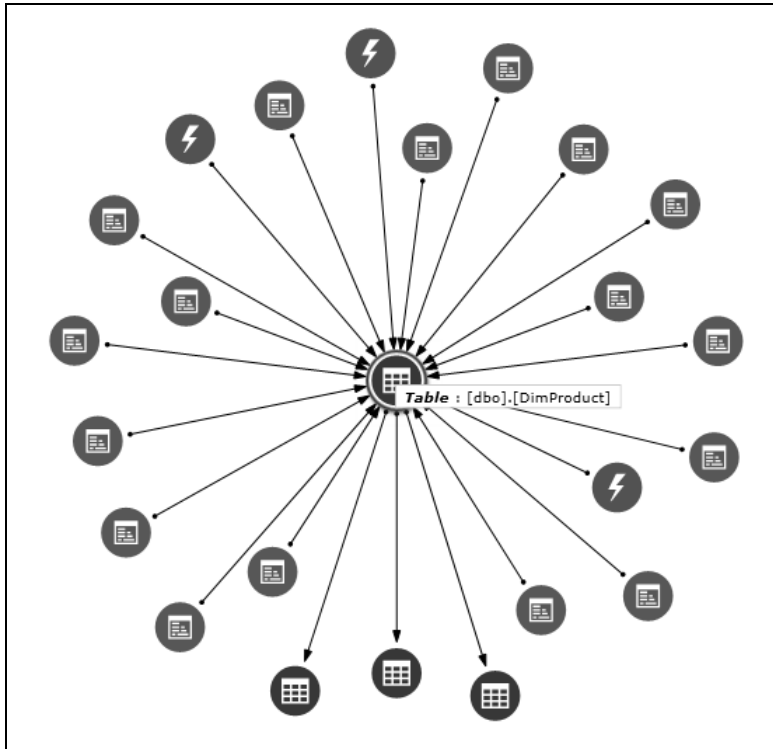


Abbildung 6.9 Ansicht verbundener Elemente der AdventureWorks2012-Tabelle *DimProduct*

Die Oberfläche visualisiert Ihnen sternförmig angeordnet sämtliche Elemente, mit denen die Tabelle *DimProduct* verbunden ist. Dazu gehören andere Tabellen, Trigger, gespeicherte Prozeduren und Sichten. Wenn Sie nun auf eines dieser Objekte klicken, ändert sich die Ansicht und wird neu auf das angeklickte Objekt ausgerichtet. Auf diese Art und Weise können Sie sich durch Ihre Datenbank klicken, sofern es Fremdschlüsselbeziehungen zwischen den Tabellen gibt. Am Anfang mag die Ansicht der vielen Elemente insbesondere in größeren Datenbanken etwas verwirrend sein. Man gewöhnt sich aber schnell an die Symbolik mit den verschiedenen Farben und lernt daher auch bald die Ansicht über *Show all dependencies* schätzen:

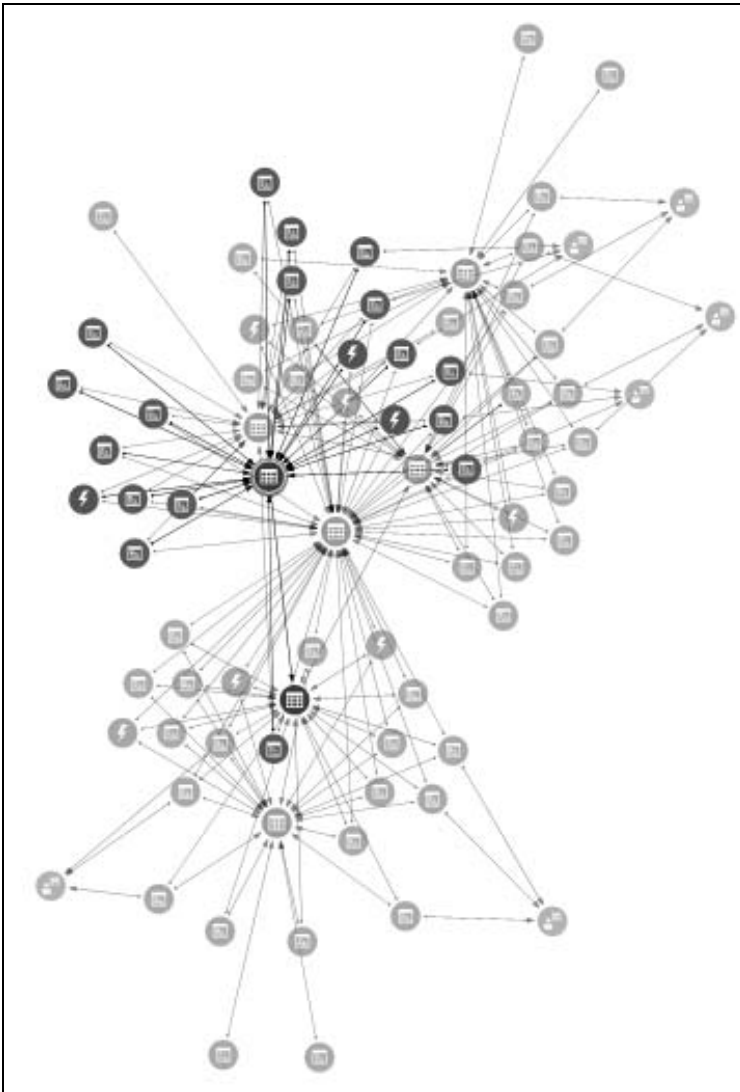


Abbildung 6.10 Überblick über das Product-Schema im Windows Azure Management Portal

Sie sehen hier alle Elemente der Tabelle *DimProduct* markiert, während alle anderen Elemente etwas abgeblendet sind. Auf diese Art und Weise können Sie sich schnell einen Überblick über alle Beziehungen in der Datenbank verschaffen.

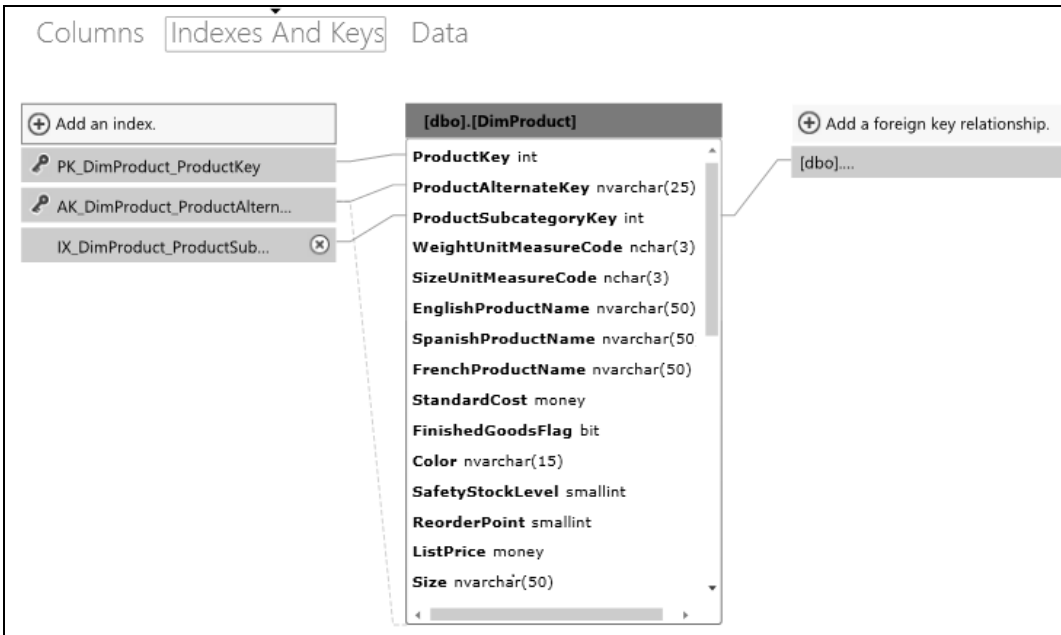


Abbildung 6.11 Referenzieren von Fremdschlüsseln im Windows Azure Management Portal

Wenn Sie einen Schritt zurückgehen und das Bearbeiten einer Tabelle aufrufen, können Sie dort Ihre Indexe und Schlüssel verwalten. Das Schöne hierbei ist die grafisch korrekte Darstellung der Beziehungen. So werden die Beziehungen zwischen den Tabellen exakt an den Spalten dargestellt, auf die sie sich beziehen.

Migrieren einer lokalen SQL Server-Datenbank

Um eine vorhandene lokale SQL Datenbank in die Wolke zu übertragen, stehen Ihnen verschiedene Möglichkeiten zu Verfügung:

- Das Einspielen einer Datenebenen-Anwendung
- Export aus dem SQL Server Management Studio
- Der Import/Export-Assistent des Managements Studios, also Integration Services
- Migrieren mittels eines SQL Server Data Tools (SSDT)

Mit SQL Server 2008 R2 wurde die Datenebenen-Anwendung eingeführt. Seitdem lässt sich im Management Studio aus einer vorhandenen Datenbank oder auch aus einem Visual Studio 2010-Datenbankprojekt eine Datenebenenanwendung extrahieren. Eine Datenebenenanwendung ist eine verpackte Form von Datenbankobjekten, zusammengefasst in einer Datei mit der Endung DACPAC. Das Transportieren einer Datenebenenanwendung unterscheidet sich von dem Anlegen eines Backups nicht nur dadurch, dass keine Daten vorhanden sind, sondern auch, dass nicht die physische Struktur (Seiten und Header, Verbindungen zur Masterdatenbank) ausgegeben wird. Datenebenen-Anwendungen können auch für den Transport von Aktualisierungen genutzt werden, beispielsweise wenn Sie in einer Entwicklungsumgebung einen neuen Stand Ihrer Datenbank entwickelt haben und diese dann auf ein Test- oder Produktivsystem verteilen möchten.

Datenebenenanwendungen lassen sich auch mit Daten erstellen. Sie erkennen eine solche Datei an der Endung BACPAC.

Der Einsatz einer Datenebenenanwendung ist geeignet, wenn Sie beispielsweise aus Sicherheitsgründen keine direkte Verbindung zwischen dem Quellserver und der Windows Azure SQL-Datenbank herstellen können.

Wenn Sie mit dem Management Studio direkten Zugriff auf Quell- und Zieldatenbank haben, ist auch der direkte Export möglich. Wählen Sie dazu nach einem Klick mit der rechten Maustaste im Punkt *Tasks* den Eintrag *Datenbank auf SQL Azure bereitstellen*. Es öffnet sich ein Wizard, der von Ihnen die Verbindungsinformationen zur Windows Azure SQL-Datenbank sowie Pfad für die temporäre Ablage einer BACPAC-Datei anfordert. Danach wird die Datenbank exportiert, validiert und auf Windows Azure bereitgestellt. Der Validierungsprozess prüft dabei, ob die Datenbank Funktionen verwendet, die in Windows Azure nicht verfügbar sind. Passiert dies, erhalten Sie die folgende Fehlermeldung.

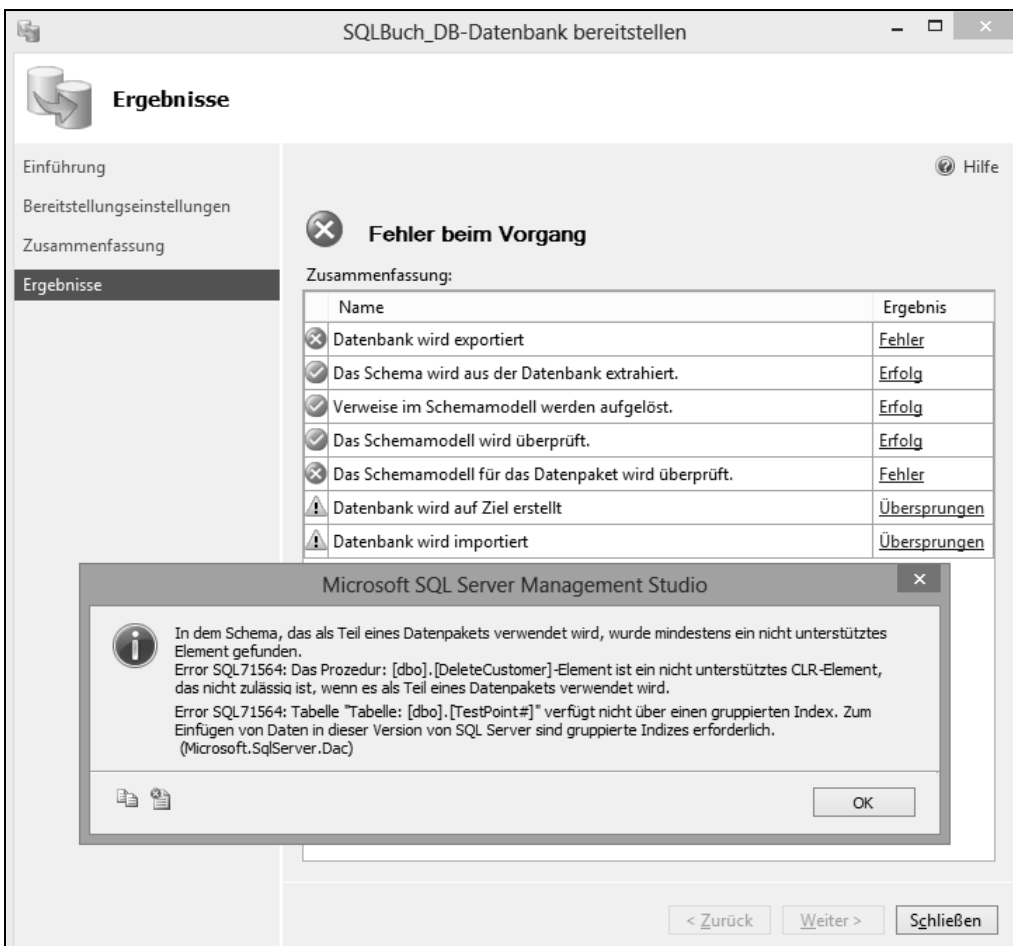
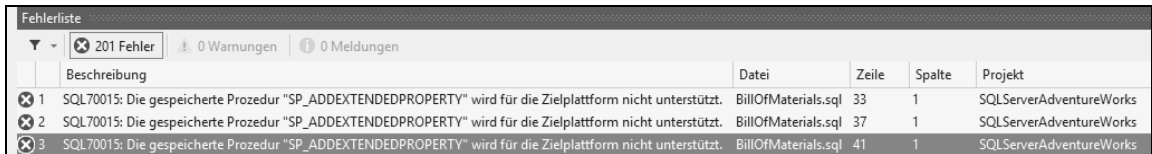


Abbildung 6.12 Fehlermeldung bei dem Versuch, eine nicht-kompatible Datenbank in Windows Azure zu transferieren

In diesem Beispiel haben wir versucht, eine Datenbank mit einer .NET CLR-Prozedur sowie einer Tabelle ohne gruppierten Index zu verwenden.

Beide bisher genannten Methoden haben den Nachteil, dass Sie erst bei dem Kopieren der Datenbankstruktur durch einen Fehler erfahren, ob Ihre vorhandene SQL Server-Datenbank überhaupt für den Einsatz auf einer Windows Azure SQL-Datenbank kompatibel ist. Die SSDT können Sie hier unterstützen. Wenn Sie ein Datenbankprojekt mit den SSDT, sei es nun mit Visual Studio 2010 oder Visual Studio 2012, angelegt haben, öffnen Sie die Eigenschaften des Projekts und stellen Sie die Zielplattform auf *Windows Azure*. Danach prüft Visual Studio umgehend die Kompatibilität und listet jeden einzelnen Verstoß als Fehler auf.



| | Beschreibung | Datei | Zeile | Spalte | Projekt |
|---|--|---------------------|-------|--------|-------------------------|
| 1 | SQL70015: Die gespeicherte Prozedur "SP_ADDEXTENDEDPROPERTY" wird für die Zielplattform nicht unterstützt. | BillOfMaterials.sql | 33 | 1 | SQLServerAdventureWorks |
| 2 | SQL70015: Die gespeicherte Prozedur "SP_ADDEXTENDEDPROPERTY" wird für die Zielplattform nicht unterstützt. | BillOfMaterials.sql | 37 | 1 | SQLServerAdventureWorks |
| 3 | SQL70015: Die gespeicherte Prozedur "SP_ADDEXTENDEDPROPERTY" wird für die Zielplattform nicht unterstützt. | BillOfMaterials.sql | 41 | 1 | SQLServerAdventureWorks |

Abbildung 6.13 Kompatibilitätsprobleme beim Überprüfen der *AdventureWorks2012*-Datenbank

Sie können nun bei jeden dieser Einträge per Doppelklick zu dem verursachenden Codesegment gelangen und eine Korrektur vornehmen, ganz ohne einen Bereitstellungs- oder Validierungsprozess anzustoßen.

HINWEIS

Microsoft aktualisiert die SQL Server Data Tools regelmäßig. Haben Sie eine Edition von SQL Server 2012 mit den SSDT oder Visual Studio 2012 installiert, empfehlen wir Ihnen unbedingt, unter <http://msdn.microsoft.com/en-us/data/hh297027> die neueste Version der SSDT abzurufen um sicher zu gehen, dass Ihr Visual Studio auch mit den aktuellen Entwicklungen bei Windows Azure versorgt ist. Auf dieser Downloadseite bietet Microsoft auch die SSDT *Power Tools* an, mit denen Sie unter anderem DACPAC-Dateien mit Daten erstellen können.

Partitionierung und Database Federation

Eine der zentralen Funktionen der Enterprise Edition von SQL Server ist die Partitionierung, mit der die Performance von Anwendungen, aber auch die Wartbarkeit und das Lockverhalten von Transaktionen Verbesserungen erfahren. Da es in einer Windows Azure SQL-Datenbank keine Möglichkeit gibt, die Datendateien und Dateigruppen selbst festzulegen, ist es auch nicht möglich, darauf basierend Partitionierungsfunktionen und Partitions-Schemata zu definieren.

Um auf einer Windows Azure SQL-Datenbank Daten zu partitionieren, ist ein wenig Umdenken erforderlich. In Windows Azure können Sie Daten über mehrere Datenbanken partitionieren, dieses Verfahren wird *vertikale Partitionierung* genannt. Wenn Sie ihre Daten über mehrere Datenbanken verteilen, können Sie nicht nur mehr als 150 GB in Windows Azure ablegen, sondern profitieren auch noch von einem Performance-Gewinn. Ihre Datenbanken werden innerhalb des Microsoft-Rechenzentrums über mehrere Server verteilt, so stehen Ihnen dadurch auch mehr Prozessor- und Arbeitsspeicherressourcen zur Verfügung. Beachten Sie beim Entwerfen einer solchen Lösung aber unbedingt, dass Sie zwar insgesamt sehr große Datenmengen speichern können, jedoch die Größe der Daten pro Partition (bzw. Datenbank) die Größe der einzelnen Datenbank, also aktuell maximal 150 GB, nicht überschreiten darf.

Dieses Vorgehen erfordert auch Anpassungen Ihres Codes. Während in der SQL Server Enterprise Edition die Aktivierung im Hintergrund erfolgt und keine Anpassungen an die Abfragen erforderlich sind, müssen Sie selbst gewährleisten, dass Ihre Abfragen die verschiedenen Datenbanken berücksichtigen. Insbesondere müssen Sie sicherstellen, dass beim Anlegen eines neuen Eintrags in einer Datenbank keine Primärschlüssel verwendet werden, die in einer anderen Datenbank bereits angelegt worden sind. Eine mögliche Lösung für dieses Problem kann das Verwenden einer .NET-Mittelschicht sein, die sich um die Verteilung der Daten auf die verschiedenen Datenbanken kümmert. Diesen Ansatz hat Wayne Walter Berry in einem Blogbeitrag beschrieben (<http://blogs.msdn.com/b/sqlazure/archive/2010/06/24/10029719.aspx>).

HINWEIS Auch bei der vertikalen Partitionierung gibt es eine Beschränkung bei der maximalen Datenmenge, die Sie in Windows Azure hinterlegen, da Sie maximal 150 Datenbanken, inklusive der Masterdatenbank, also maximal 149 Benutzerdatenbanken, anlegen können.

Seit Kurzem bietet Microsoft noch einen weiteren Ansatz an, große Datenmengen in Windows Azure SQL-Datenbanken über eine so genannte *Federation* zu verteilen. Eine Federation erinnert schon ein wenig mehr an die Partitionierung aus dem SQL Server, ist jedoch auch eine vertikale Partitionierung. So gilt es bei der Verwendung, ein Federation-Schema anzulegen, welches einen *Verteilungsschlüssel* vom Typ INT, BIGINT, UNIQUEIDENTIFIER oder VARBINARY (bis zu 900 Bytes) enthalten kann. Anhand des Verteilungsschlüssels wird entschieden, auf welche Partitionen die jeweiligen Daten aufgeteilt werden. Die Partitionen heißen in diesem Kontext *Federation Members* und sind auch hier als Datenbanken implementiert.

Innerhalb einer Federation haben Sie die Möglichkeit, Federation Tables anzulegen, die dann Ihre Daten auf die verschiedenen Mitglieder verteilen. Ebenso können Sie Referenztabellen anlegen, die nicht über die Federation Members aufgeteilt werden. Um die Sammlung der Begrifflichkeiten komplett zu machen, gibt es auch noch eine Bezeichnung für die Zusammenfassung aller Datensätze, die sich in einem Federation Member befinden. Diese Datensätze verfügen alle über denselben Verteilungsschlüssel und werden so dem Federation Member zugeordnet, die Gruppe dieser Datensätze heißt *Federation Atomic Unit*.

Sie können innerhalb jedes Mitgliedes (also jeder Datenbank) aber auch Referenztabellen anlegen, diese werden nicht aufgeteilt, sondern stellen eine lokale Tabelle dar. Die Möglichkeit, Referenztabellen anzulegen, wurde geschaffen, damit Sie in jedem Federation Member lokale Informationen zu den Federation Tables hinterlegen können, aber auch, da Federation Tables wieder einer ganzen Reihe von Einschränkungen unterliegen.

Gegenwärtig sind in Federation Tables unter anderem keine berechneten Spalten, indizierte Sichten und NULL-Werte enthalten. Eine Liste der genauen Beschränkungen finden Sie in MSDN unter <http://msdn.microsoft.com/en-us/library/windowsazure/hh597469.aspx>.

Der wirklich große Vorteil beim Einsatz einer Federation besteht darin, dass Ihnen eine *Federation Root-Datenbank* angeboten wird. Diese Datenbank ist der Verbindungspunkt für alle Anwendungen, die mit der Federation arbeiten möchten. Für die Anwendung, oder auch ETL-Prozesse (Extrahieren, Transformieren und Laden), ist somit die Logik der Federation, genauso wie bei der SQL Server-Partitionierung, wieder versteckt und erfordert keine umfangreichen Anpassungen. Die Federation Root-Datenbank kümmert sich darum, die Datenbankabfragen auf die jeweiligen Federation Member zu leiten und neue Datensätze entsprechend aufzuteilen.

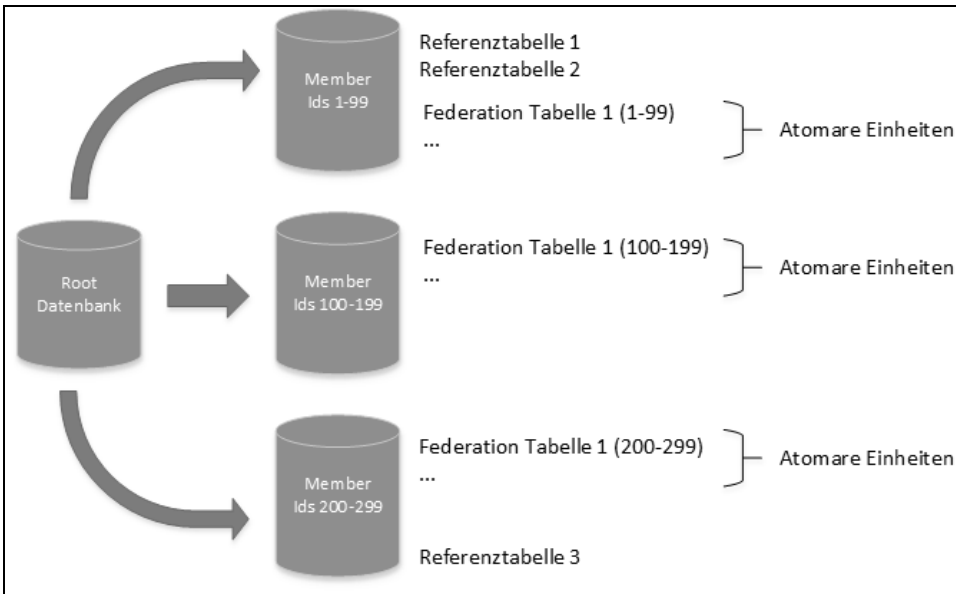


Abbildung 6.14 Architektur der Windows Azure SQL-Datenbank Federation

SQL Reporting

Neben der Möglichkeit, Daten in der Wolke abzulegen, können Sie in Windows Azure auch SQL Server Reports bereitstellen. Dafür stellt Microsoft das *Windows Azure SQL Reporting* bereit, sowie die Möglichkeit, Berichte in einer Webanwendung über das ReportViewer-Steuerelement ohne den Einsatz eines Berichtsservers zu rendern.

HINWEIS Die aktuelle Version des Windows Azure SQL Reporting basiert auf Reporting Services 2008 R2, weshalb Sie gegenwärtig Berichte noch mit Visual Studio 2008 bzw. der Visual Studio Shell des SQL Server 2008 R2 erstellen müssen. Microsoft stellt soweit alle Funktionen der Enterprise Edition von Reporting Services zur Verfügung, sogar zeitgesteuerte datengetriebene Abonnements, die per E-Mail verteilt werden können. Die einzige Einschränkung ist das Fehlen der Erweiterbarkeit. Sie können in einem *Windows Azure SQL Reporting*-Server keine Assemblys registrieren und diese somit auch nicht in Ihren Berichten referenzieren.

Um einen Bericht in Windows Azure SQL Reporting anzeigen zu können, müssen Sie in der Verwaltungskonsolle zuerst eine Berichtsserverinstanz erstellen. Klicken Sie dazu im Abschnitt *Server* auf den Menüpunkt *Create*. Es wird ein Wizard gestartet. Es wird ein Assistent gestartet, die Sie wie üblich danach fragt, wo Ihr Server geografisch abgelegt werden soll. Nach der Vergabe eines Administratorkennworts können Sie über den Menüpunkt *Manage* Benutzer anlegen und diesen Rollen zuweisen.

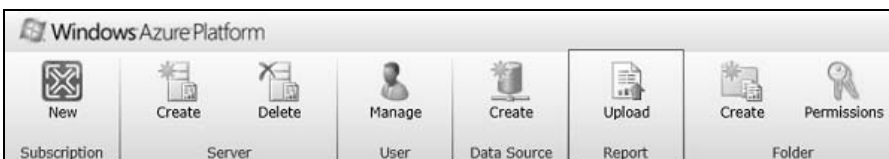


Abbildung 6.15 Menüpunkt zum Hochladen von Berichten in Windows Azure SQL Reporting

Ist der Reportserver betriebsbereit, können Sie über den Menüpunkt *Upload* eine *.rdl*-Datei auswählen, dem Bericht einen Namen geben und das Paket dann hochladen. Nun benötigen Sie noch eine Datenquelle, diese können Sie erstellen, indem Sie im Abschnitt *Data Source* auf *Create* klicken, dort können Sie Ihre Verbindungszeichenfolge zu Ihrer Windows Azure SQL-Datenbank hinterlegen.

Abbildung 6.16 Anlegen einer Datenquelle in Windows Azure SQL Reporting

Die Verbindungszeichenfolge kann hier leider nicht über ein Dialogfeld erstellt werden. Halten Sie sich an das folgende Format, um die Verbindung anzugeben:

```
Data Source= <SQL Database service>; Initial Catalog= <SQL Database instance>; Encrypt=True;
```

Listing 6.2 Verbindungszeichenfolge zu einer Windows Azure SQL-Datenbank

Die Funktionalität des Management Portals ähnelt sehr dem Berichtsmanager, in Reporting Services. Sofern der Bericht über eine lokale Datenverbindung zu Ihrer Windows Azure SQL-Datenbank verfügt, können Sie den Bericht im Anschluss dann rendern.

Dazu benötigen Sie die URL, unter der Ihr Bericht erreichbar ist. In der Mitte des Verwaltungsportals wird Ihnen die URL Ihres Berichtsservers im Abschnitt *Web Service URL* angezeigt. Klicken Sie dort auf den Link, um auf den Berichtsserver zu gelangen. Sie erhalten die gleiche Ansicht wie bei einem lokal installierten Berichtsserver und können hier die Verzeichnisse Ihres Berichtsservers durchsuchen. Klicken Sie nun einfach auf den Namen eines Berichts, um ihn zu starten. Die nun in Ihrem Browser angezeigte Adresse können Sie einfach an die Benutzer weitergeben, damit diese Ihren Bericht direkt erreichen können. Diese Adresse hat typischerweise das folgende Format:

`http://<ihrserver>.report.int.mscds.com/ReportServer/<Ordner>/<Bericht>.rdl`

Listing 6.3 Typische URL zu einem Bericht

Wenn Sie einen Bericht über Windows Azure öffnen, unterscheidet sich dieser optisch nicht von einem Bericht, den Sie in Reporting Services rendern lassen. Die folgende Abbildung zeigt einen einfachen Product Sales KPI-Bericht (Key Performance Indicator), der in Windows Azure bereitgestellt worden ist.

| Sales Date | Subcategory1 | Product | Quantity | Sales | KPI Gauge | KPI |
|------------|--------------|---------------------------|------------|-------------|-----------|-----|
| 1/5/2009 | Accessories | Carrying Case | 68 | \$16,996.60 | | |
| | | Mini Battery Charger | 44 | \$1,056.00 | | |
| | | | 112 | \$18,052.60 | | |
| | Digital | Slim Digital | 44 | \$8,357.80 | | |
| | | | 44 | \$8,357.80 | | |
| | Total | | 156 | \$26,410.40 | | |
| 1/6/2009 | Accessories | Telephoto Conversion Lens | 18 | \$1,380.00 | | |
| | | Tripod | 18 | \$1,350.00 | | |
| | | USB Cable | 26 | \$780.00 | | |
| | | | 62 | \$3,510.00 | | |
| | Total | | 62 | \$3,510.00 | | |
| 1/7/2009 | Digital | Compact Digital | 84 | \$10,836.00 | | |
| | | | 84 | \$10,836.00 | | |
| | Digital SLR | SLR Camera | 88 | \$26,576.00 | | |
| | | | 88 | \$26,576.00 | | |
| | Total | | 172 | \$37,412.00 | | |
| 1/8/2009 | Accessories | Budget Movie-Maker | 9 | \$3,798.00 | | |
| | | | 9 | \$3,798.00 | | |
| | Digital | Consumer Digital | 17 | \$2,550.00 | | |
| | | 17 | \$2,550.00 | | | |
| | Total | | 26 | \$6,348.00 | | |

Abbildung 6.17 Beispiel für einen Bericht in Windows Azure SQL Reporting

Wenn Sie mehrere Berichte in einem Projekt entwickeln, ist das Hochladen über das Verwaltungportal zu mühselig und kann zu einer zeitraubenden Zeitarbeit werden. Sie können Ihre Berichte auch direkt aus Visual Studio bereitstellen, indem Sie die Adresse Ihres Berichtsservers in der Eigenschaft *TargetServerURL*

Ihres Berichtsprojekts hinterlegen. Wenn wir uns an dem Beispiel aus Listing 6.3 orientieren, tragen Sie die Adresse `http://<ihrserver>.report.int.mscds.com/ReportServer` ein. Nun können Sie mit einem Rechtsklick auf Ihr Berichtsprojekt einfach den Eintrag *Bereitstellen* wählen und Ihre Berichte werden alle innerhalb kürzester Zeit aktualisiert.

HINWEIS Bei Erscheinen des Buchs war das Windows Azure SQL Reporting leider noch nicht in dem neuen Verwaltungsportal verfügbar. Es ist durchaus möglich, dass Microsoft im Zuge der Migration dorthin auch die aktuelle Version der Reporting Services unterstützen wird. Schauen Sie am besten selbst direkt rein.

Office 365: Business Intelligence mit SharePoint Online

Microsoft bietet mit Office 365 eine ganze Reihe von Diensten als *Software as a Service (SaaS)* an. Dazu zählt unter anderen Exchange Online, Lync Online, Dynamics CRM Online, aber auch: SharePoint Online.

Im November 2012 plant Microsoft, eine Aktualisierung der Office 365-Dienste bereitzustellen, und alles deutet daraufhin, dass die neue Version 2013 von Office, SharePoint und Exchange Einzug in die Online-dienste halten wird. Es ist bereits angekündigt, dass es zukünftig möglich sein wird, PowerPivot-Tabellen und auch Power View-Berichte in SharePoint Online zu nutzen.

Somit können Sie neben Berichten auf Ihrer relationalen Windows Azure SQL-Datenbank auch das neue xVelocity Modell nutzen, um Berichte via Excel Web App und Power View anzuzeigen.

Nun, wo Sie dieses Buch in den Händen halten, müsste die Aktualisierung bereits verfügbar sein. Werfen Sie daher am besten direkt einen Blick auf die aktuellen Office 365-Angebote.

Zusammenfassung

Die Datenbank in der Wolke ist inzwischen zu einem nutzbaren Produkt geworden. Für die Beschränkungen hinsichtlich ablegbarer Dateigröße gibt es Lösungen und auch die Migration nach Windows Azure ist mit den neuen Data Tools wirklich einfach geworden. Es bleibt abzuwarten, welche Funktionen als nächstes eingebaut werden und somit immer mehr Datenbanken ohne Umbau direkt von Windows Azure unterstützt werden.

Die Entwickler des Windows Azure Management Portals haben einiges geleistet, um die Verwaltung der Datenbank einfach zu gestalten und so sind Features entstanden, wie die Übersicht der Fremdschlüsselbeziehungen, die man sich sicher auch gut im Management Studio vorstellen könnte.

Mit der Integration von Hadoop in Windows Azure zeigt Microsoft, dass quelloffene Software, sofern Sie innovativ ist, durchaus auch für Microsoft interessant sein kann. Übrigens soll auch MySQL künftig in Windows Azure angeboten werden.

Es bleibt also spannend, wie Microsoft das Reporting in Zukunft gestalten wird und welche Möglichkeiten durch die neuen Excel Web Apps entstehen werden. Sicher wird es auch dort architekturbedingt Einschränkungen geben, wir sind aber zuversichtlich, dass die Lösung eine Bereicherung für die Bereitstellung von Berichten sein wird.

Das nächste Kapitel bringt Ihnen die Integration Services von SQL Server 2012 etwas näher. Auf dass die Datensätze gehorchen mögen.

