

SQL Azure Die Datenbank in der Cloud

Rainer Stropek
software architects gmbh
MVP für Windows Azure

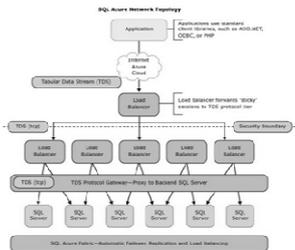
Abstract

Die Windows Azure Plattform bietet uns als Datenbankentwickler Datenbankcluster zwischen 1 und 150 GB, die in wenigen Sekunden angelegt und nach Nutzung abgerechnet werden. Backup, Import/Exportdienste, mehrstufige Firewalls u.w.m. sind im Paket enthalten und brauchen weder extra bezahlt noch konfiguriert noch gewartet werden.

- Überblick über die Architektur von SQL Azure
- Mechanismen zum Datenschutz der DB-Server in der Cloud
- Unterschiede zwischen On-premise SQL Server und SQL Azure
- Welche T-SQL Kommandos gibt es in der Cloud nicht, welche kommen dazu
- Ein Beispiel zeigt, wie einfach von .NET- und Windowsanwendungen auf SQL Azure zugegriffen werden kann

Windows Azure

SQL Azure Database Architecture (As of October 2011)



Windows Azure

SQL Azure Billing Rates



Web Edition
100 MB up to 5 GB

Business Edition
5 GB up to 150 GB

Pricing
 \$4.995 per database up to 100 MB per month
 \$9.99 per database >100 MB up to 1 GB per month
 \$9.99 for first GB, \$3.996 for each additional GB per database >1 GB up to 10 GB per month
 \$45.954 for first 10 GB, \$1.998 for each additional GB per database >10 GB up to 50 GB per month
 \$125.874 for first 50 GB, \$0.999 for each additional GB per database >50 GB up to 150 GB per month

[Windows Azure Pricing Calculator](#)

Windows Azure

Starting with the Basics

Start With the Basics



SQL Azure Database is
 SQL Server database technology delivered as a service on the Windows Azure Platform
 Ideal for both simple and complex applications
 Enterprise-ready with automatic support for HA
 Designed to scale out elastically with demand

Get started quickly
 Choose a plan
 Choose a billing option
 Provision servers

Windows Azure

Creating a SQL Azure DB

demo

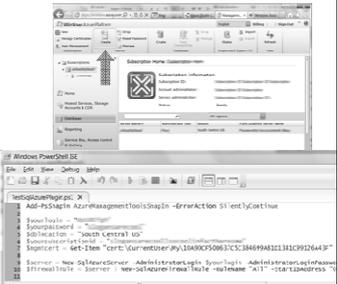


Provision Your Server

Server defined
 Service head that contains databases
 Connect via automatically generated FQDN (xxx.database.windows.net)
 Initially contains only a **master** database

Provision servers interactively
 Log on to Windows Azure Management Portal
 Create a SQL Azure server
 Specify admin login credentials
 Add firewall rules and enable service access

Automate server provisioning
 Use Windows Azure Platform PowerShell cmdlets (or use REST API directly)
wappowershell.codeplex.com



Build and Deploy your Database

Management Studio
DAC Deployment



demo

Build Your Database

| | |
|--|---|
| <p>Use familiar technologies</p> <ul style="list-style-type: none"> Supports Transact-SQL Supports popular languages <ul style="list-style-type: none"> .NET Framework (C#, Visual Basic, F#) via ADO.NET C / C++ via ODBC Java via Microsoft JDBC provider PHP via Microsoft PHP provider Supports popular frameworks <ul style="list-style-type: none"> OData (REST data access) Entity Framework WCF Data Services NHibernate Supports popular tools <ul style="list-style-type: none"> SQL Server Management Studio (2008 R2 and later) SQL Server command-line utilities (SQLCMD, BCP) CA Erwin® Data Modeler Embarcadero Technologies DBArtisan® <p><small>Windows Azure</small></p> | <p>Differences in comparison to SQL Server</p> <ul style="list-style-type: none"> Focus on logical vs. physical administration Database and log files automatically placed Three high-availability replicas maintained for every database Databases are fully contained Tables require a clustered index Maximum database size is 50 Gb <p>Unsupported SQL Server features</p> <ul style="list-style-type: none"> BACKUP / RESTORE USE command, linked servers, distributed transactions, distributed views, distributed queries, four-part names Service Broker Common Language Runtime (CLR) SQL Agent |
|--|---|

Build Your Database



| |
|---|
| <p>Thin client database development</p> <ul style="list-style-type: none"> Introducing the SQL Azure management portal Web designers for tables, views, stored procs Interactive query editing and execution <p>Rich client database development</p> <ul style="list-style-type: none"> Introducing SQL Server Data Tools (SSDT) Visual Studio IDE for database development Includes modern designers and projects with declarative, model-driven development Develop and test in both connected and disconnected states Platform targeting for both SQL Server (2005 and above) and SQL Azure Get it free with Web PI, with SQL Server 2012 and with Visual Studio 11 <p><small>Windows Azure</small></p> |
|---|

Deploy Your Database



Data-tier Application Framework (DAC Fx)
 Alternative to traditional script based approach
 Dramatically simplifies deployment, migration and versioning of databases
 Provides a single unit of deployment for schema (dacpac) or for schema + data (bacpac)
 Supports automatic versioning of database schemas
 Supports platform targeting for both SQL Server (2005 and above) and SQL Azure
 Build from scratch or extract from existing db

How to get the latest DAC Fx
 With SQL Server Data Tools
 With SQL Server 2012 Management Studio
 With SQL Azure Import/Export Service
 Via sqldacexamples.codeplex.com

WindowsAzure

Deploy Your Database



Interactive approach for dacpac v1 and v2
 Manage your server with SQL Azure Management Portal
 Deploy dacpac
 Load data (scripts, bulk copy, SSIS)
 Also supported in SQL Server 2012 Management Studio



Interactive approach for bacpac v2
 Upload bacpac to blob storage
 Log into Windows Azure Management Portal
 Import bacpac, table data automatically loaded
 Also supports export to blob storage

Upgrading a dacpac or bacpac
 Not supported in portals yet
 Use SQL Server 2012 Management Studio

WindowsAzure

Secure your Database

Access SQL Azure
from .NET

demo



There are two ways to secure a database:



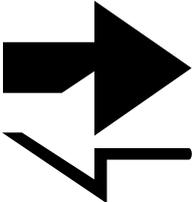
On the Server



Within the Database

Windows Azure

Benefits



Server identity and access control

- SQL authentication supported
- Integrated authentication not supported
- Connect to **master** to administer logins and create / drop databases
- The admin login (configured during service provisioning) is like **sa**
- The admin login has full rights on the server (and all databases) and should only be used for administration
- Manage logins with **CREATE / ALTER / DROP LOGIN** commands
- Membership in the **loginmanager** server role grants **CREATE / ALTER / DROP LOGIN** privileges
- Membership in the **dbmanager** server role grants **CREATE / DROP DATABASE** privileges

Windows Azure

Benefits

Database identity and access control
 Logins must have an associated user account to connect to a database
 The admin login is automatically associated with a special user known as **dbo** (database owner)
 The dbo has full rights in the database and should only be used for administration
 Manage users with CREATE / ALTER / DROP USER commands
 Add users to system or user-defined database roles to grant privileges via sp_add_rolemember
 Organize database objects into schema containers based upon common access control requirements
 Grant privileges to schema containers instead of individual objects for better productivity



Windows Azure

Connect Your Application

Connecting to SQL Azure
 TDS (Tabular Data Stream) protocol over TCP/IP supported
 SSL required
 Use firewall rules to connect from outside Microsoft data center
 ASP.NET example:

```
<connectionStrings>
  <add name="AdventureWorks" connectionString="
    Data Source=[server];database=windows.net;
    Integrated Security=false;
    Initial Catalog=Productsdb;
    User Id=[login];
    Password=[password];
    Encrypt=true"
  />
  <providerName="System.Data.SqlClient"/>
</connectionStrings>
```

Special considerations
 Legacy tools and providers may require special format for login: [login]@[server]
 Idle connections terminated after 30 minutes
 Long running transactions terminated after 24 hours
 DoS guard terminates suspect connections with no error message
 Failover events terminate connections
 Throttling may cause errors
 Use connection pooling and implement retry logic to handle transient failures
 Latency introduced for updates due to HA replicas
 No cross-database dependencies, result sets from different databases must be combined in application tier

Windows Azure

Exploring advanced Capabilities

SQL Azure
DataSync

demo



Visualize Your Data

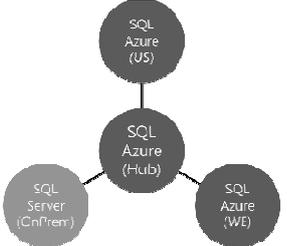


SQL Azure Reporting is
 SQL Server Reporting Services technology delivered as a service on the Windows Azure Platform
 Ideal for operational reporting against SQL Azure data
 Enterprise-ready with automatic support for HA
 Designed to scale elastically with demand

Get started quickly
 Provision report server via Windows Azure Management Portal
 Build reports with Reporting Services Report Designer
 Deploy reports to report server
 Render reports with Visual Studio ReportViewer controls

Windows Azure

Synchronize Your Data



SQL Azure Data Sync is
 Microsoft Sync Framework technology delivered as a service on the Windows Azure Platform
 Ideal for scheduling synchronization between data sets hosted in SQL Azure or SQL Server
 Uses a hub and spoke topology

Special considerations
 Conflict resolution policy configured centrally (hub or client wins)
 Sync direction configured between each client and the hub (to hub, from hub, bi-directional)
 Sync schedule must be between 5 minutes and 1 month
 Data sets include multiple tables and can be filtered, triggers are added to data set tables
 Tables added to hub and client schemas
 Agent must be installed for on-prem clients

Windows Azure

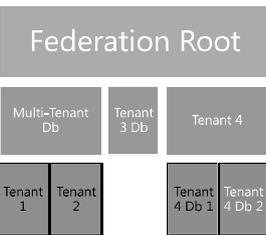
Synchronize Your Data



- Provisioning synchronization groups
 - Deploy database to hub and clients
 - Set synchronization schedule
 - Set conflict resolution policy
 - Define data set
 - Add SQL Azure clients to topology
 - Install agents on SQL Server clients and add them to topology using keys
 - Deploy topology
- Operational considerations
 - Manual synchronization supported
 - Hub updates require one synchronization to distribute to all clients
 - Client updates require two synchronizations to distribute to all clients
 - Sync Framework triggers may affect application behavior

WindowsAzure

Scale Out Your Data



SQL Azure Federations provides

- Integrated database sharding that can scale to hundreds of nodes
- Multi-tenancy via flexible repartitioning
- Online split operations to minimize downtime
- Automatic data discovery regardless of changes in how data is partitioned

Special considerations

- A logical database can contain multiple federations
- Distribution scheme supports int, bigint, guid, and varbinary types
- Filtering routes connection to appropriate shard regardless of changes in partitions
- Merge, fan-out queries and automatic distribution of schema changes not supported in initial release

WindowsAzure

SQL Azure Federations

demo



Vielen Dank für die
Aufmerksamkeit!
