



SQL Server on Linux

Eine fixe Idee wird Realität

About Me



www.sql-aus-hamburg.de



[@SQL_aus_HH](https://twitter.com/SQL_aus_HH)



info@sql-aus-hamburg.de



[SQL_aus_HH](https://www.instagram.com/SQL_aus_HH)



Agenda

Historie

Features / Möglichkeiten

Installation

Konfiguration

Migration/Restore

mögliche Szenarien

SQL Server vNext im Container

Fazit

Q&A

Historie

SQL Server 4.21 (erstmalig auf Windows)

SQL Server 6.00 (1995)

SQL Server 6.5 (1996)

SQL Server 8.0 (2000)

SQL Server 2005 (2005)

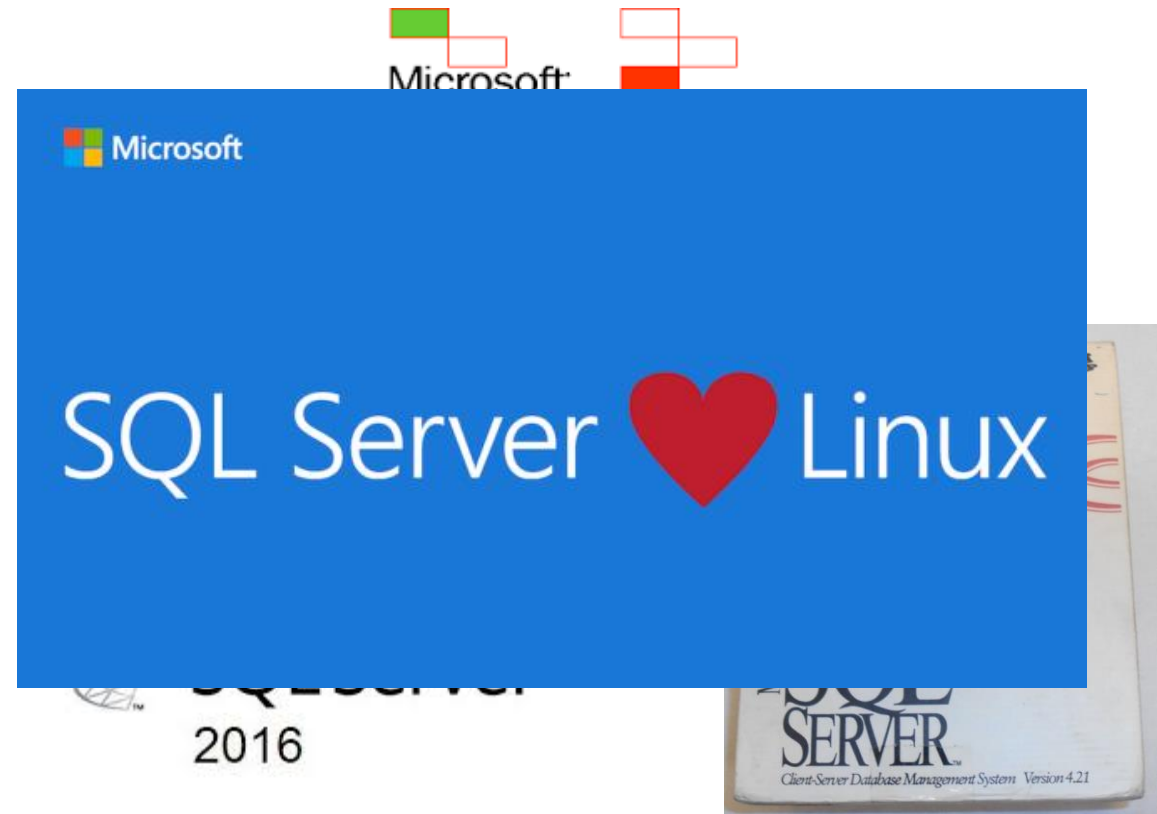
SQL Server 2008 (2008)

SQL Server 2012 (2012)

SQL Server 2014 (2014)

SQL Server 2016 (2016)

SQL Server vNext (Mitte 2017)



Features / Möglichkeiten

DATABASE ENGINE

Full-text Search
Replication
Stretch DB
Polybase
Distributed Query
System extended stored procedures (XP_CMDSHELL, etc.)
Filetable
CLR assemblies with the EXTERNAL_ACCESS or UNSAFE permission set

HIGH AVAILABILITY

Always On Availability Groups
Database mirroring

SECURITY

Active Directory Authentication
Windows Authentication
Extensible Key Management
Use of user-provided certificate for SSL or TLS

SERVICES

SQL Server Agent
SQL Server Browser
SQL Server R services
StreamInsight
Analysis Services
Reporting Services
Integration Services
Data Quality Services
Master Data Services

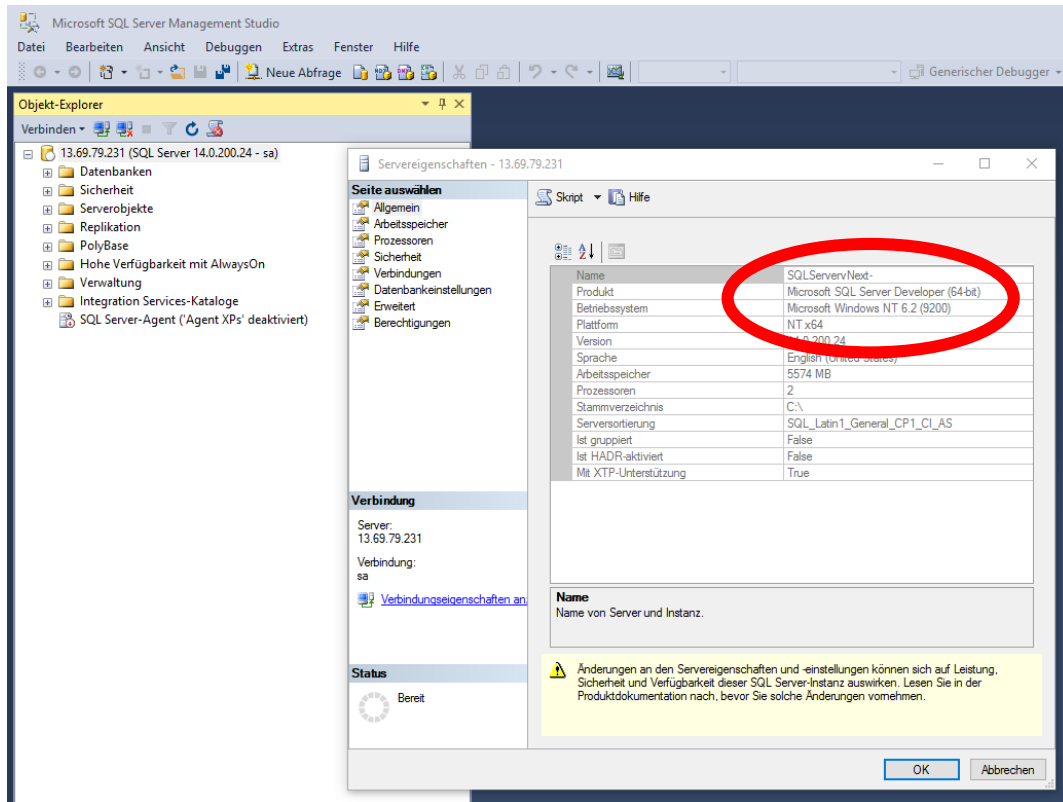
Installation

```
1. ## Add the Microsoft repository to the Ubuntu repository
2. sudo su
3. curl https://packages.microsoft.com/keys/microsoft.asc | apt-key add -
4. curl https://packages.microsoft.com/config/ubuntu/16.04/mssql-server.list > /etc/apt/sources.list.d/mssql-server.list
5. exit
6.
7. ## Install SQL
8. sudo apt-get update
9. sudo apt-get install -y mssql-server
10.
11. ## Configure SQL
12. sudo /opt/mssql/bin/sqlservr-setup
13.
14. ## Check if SQL is running
15. systemctl status mssql-server
```

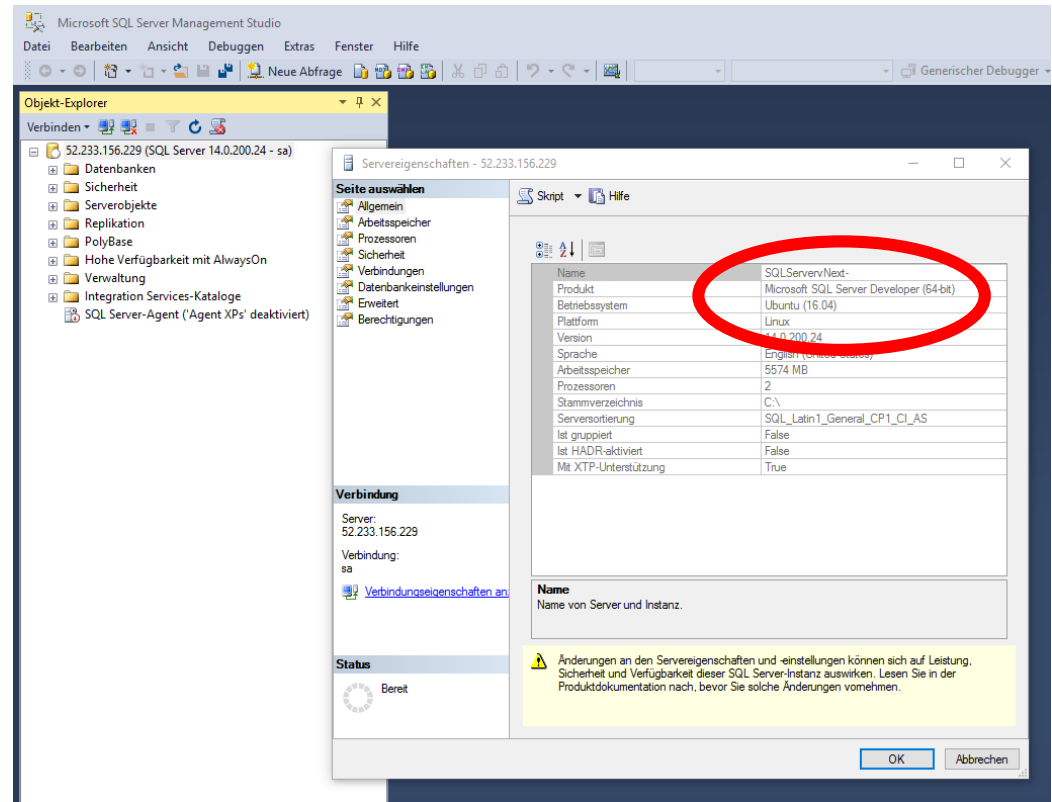
Installation

Demo-Time

Besonderheit



SSMS v16.x



SSMS v17.0 RC1

Konfiguration



Fast analog zum SQL Server 2016
Powershell, ManagementStudio, TSQL

NEU – Kommandozeile - mssql-conf

Konfiguration – mssql-conf

Erstellen neuer Daten-/Tlog-Verzeichnisse (Mountpoints/Filesysteme)

```
sudo mkdir /tmp/data  
sudo mkdir /tmp/tlog
```

Berechtigungen und Eigentümer der Verzeichnisse anpassen

```
sudo chown mssql /tmp/data  
sudo chgrp mssql /tmp/data  
sudo chown mssql /tmp/tlog  
sudo chgrp mssql /tmp/tlog
```

Default-Pfade für Data und Tlog ändern

```
sudo /opt/mssql/bin/mssql-conf set defaultdatadir /tmp/data  
sudo /opt/mssql/bin/mssql-conf set defaulttlogdir /tmp/tlog
```

Restart the SQL Server service as instructed by the configuration utility:

```
sudo systemctl restart mssql-server
```

Konfiguration – mssql-conf

Demo-Time

Migration / Restore



```
RESTORE DATABASE AdventureWorks FROM DISK =  
'/var/opt/mssql/backup/AdventureWorks2014.bak'  
WITH MOVE 'AdventureWorks2014_Data' TO  
'/tmp/data/AdventureWorks2014_Data.mdf',  
MOVE 'AdventureWorks2014_Log' TO  
'/tmp/tlog/AdventureWorks2014_Log.ldf';  
GO
```

```
1> opt/mssql/backup/AdventureWorks2014.bak' WITH MOVE 'AdventureWorks2014_Data' TO '/tm  
2> GO  
Processed 24248 pages for database 'AdventureWorks', file 'AdventureWorks2014_Data' on  
Processed 4 pages for database 'AdventureWorks', file 'AdventureWorks2014_Log' on file  
Converting database 'AdventureWorks' from version 782 to the current version 860.  
Database 'AdventureWorks' running the upgrade step from version 782 to version 801.  
Database 'AdventureWorks' running the upgrade step from version 801 to version 802.  
Database 'AdventureWorks' running the upgrade step from version 802 to version 803.  
Database 'AdventureWorks' running the upgrade step from version 803 to version 804.  
Database 'AdventureWorks' running the upgrade step from version 804 to version 805.  
Database 'AdventureWorks' running the upgrade step from version 805 to version 806.  
Database 'AdventureWorks' running the upgrade step from version 806 to version 807.  
Database 'AdventureWorks' running the upgrade step from version 807 to version 808.  
Database 'AdventureWorks' running the upgrade step from version 808 to version 809.  
Database 'AdventureWorks' running the upgrade step from version 809 to version 810.  
Database 'AdventureWorks' running the upgrade step from version 810 to version 811.  
Database 'AdventureWorks' running the upgrade step from version 811 to version 812.  
Database 'AdventureWorks' running the upgrade step from version 812 to version 813.  
Database 'AdventureWorks' running the upgrade step from version 813 to version 814.  
Database 'AdventureWorks' running the upgrade step from version 814 to version 815.  
Database 'AdventureWorks' running the upgrade step from version 815 to version 816.  
Database 'AdventureWorks' running the upgrade step from version 816 to version 817.  
Database 'AdventureWorks' running the upgrade step from version 817 to version 818.  
Database 'AdventureWorks' running the upgrade step from version 818 to version 819.  
Database 'AdventureWorks' running the upgrade step from version 819 to version 820.  
Database 'AdventureWorks' running the upgrade step from version 820 to version 821.  
Database 'AdventureWorks' running the upgrade step from version 821 to version 822.  
Database 'AdventureWorks' running the upgrade step from version 822 to version 823.  
Database 'AdventureWorks' running the upgrade step from version 823 to version 824.  
Database 'AdventureWorks' running the upgrade step from version 824 to version 825.  
Database 'AdventureWorks' running the upgrade step from version 825 to version 826.  
Database 'AdventureWorks' running the upgrade step from version 826 to version 827.  
Database 'AdventureWorks' running the upgrade step from version 827 to version 828.  
Database 'AdventureWorks' running the upgrade step from version 828 to version 829.  
Database 'AdventureWorks' running the upgrade step from version 829 to version 830.  
Database 'AdventureWorks' running the upgrade step from version 830 to version 831.  
Database 'AdventureWorks' running the upgrade step from version 831 to version 832.  
Database 'AdventureWorks' running the upgrade step from version 832 to version 833.  
Database 'AdventureWorks' running the upgrade step from version 833 to version 834.  
Database 'AdventureWorks' running the upgrade step from version 834 to version 835.  
Database 'AdventureWorks' running the upgrade step from version 835 to version 836.  
Database 'AdventureWorks' running the upgrade step from version 836 to version 837.  
Database 'AdventureWorks' running the upgrade step from version 837 to version 838.  
Database 'AdventureWorks' running the upgrade step from version 838 to version 839.  
Database 'AdventureWorks' running the upgrade step from version 839 to version 840.  
Database 'AdventureWorks' running the upgrade step from version 840 to version 841.  
Database 'AdventureWorks' running the upgrade step from version 841 to version 842.  
Database 'AdventureWorks' running the upgrade step from version 842 to version 843.  
Database 'AdventureWorks' running the upgrade step from version 843 to version 844.  
Database 'AdventureWorks' running the upgrade step from version 844 to version 845.  
Database 'AdventureWorks' running the upgrade step from version 845 to version 846.  
Database 'AdventureWorks' running the upgrade step from version 846 to version 847.  
Database 'AdventureWorks' running the upgrade step from version 847 to version 848.  
Database 'AdventureWorks' running the upgrade step from version 848 to version 849.  
Database 'AdventureWorks' running the upgrade step from version 849 to version 850.  
Database 'AdventureWorks' running the upgrade step from version 850 to version 851.  
Database 'AdventureWorks' running the upgrade step from version 851 to version 852.  
Database 'AdventureWorks' running the upgrade step from version 852 to version 853.  
Database 'AdventureWorks' running the upgrade step from version 853 to version 854.  
Database 'AdventureWorks' running the upgrade step from version 854 to version 855.  
Database 'AdventureWorks' running the upgrade step from version 855 to version 856.  
Database 'AdventureWorks' running the upgrade step from version 856 to version 857.  
Database 'AdventureWorks' running the upgrade step from version 857 to version 858.  
Database 'AdventureWorks' running the upgrade step from version 858 to version 859.  
Database 'AdventureWorks' running the upgrade step from version 859 to version 860.  
RESTORE DATABASE successfully processed 24252 pages in 5.683 seconds (33.338 MB/sec).  
1> |
```

Backup

```
sqlcmd -H localhost -U SA -Q "BACKUP DATABASE [AdventureWorks] TO DISK =  
N'/var/opt/mssql/backup/AdventureWorks2014_Backup_2017-02-09.bak' WITH NOFORMAT, NOINIT,  
NAME = N'AdventureWorks2014_Backup_2017-02-09', SKIP, NOREWIND, NOUNLOAD, STATS = 10"
```

```
sqlcmd -H localhost -U SA -Q "BACKUP LOG [AdventureWorks2014] TO DISK =  
N'var/opt/mssql/data/AdventureWorks2014_LogBackup_2017-02-09.bak' WITH NOFORMAT, NOINIT,  
NAME = N'AdventureWorks2014_LogBackup_2017-02-09', NOSKIP, NOREWIND, NOUNLOAD,  
NORECOVERY , STATS = 5"
```

```
root@SQLServervNext-2:/home/dbadmin# sudo sqlcmd -H localhost -U SA -Q "BACKUP DATABASE [AdventureWorks] TO DISK = N'/var/opt/mssql/backup/AdventureWorks2014_Backup_2017-02-09.bak'  
NOINIT, NAME = N'AdventureWorks2014_Backup_2017-02-09', SKIP, NOREWIND, NOUNLOAD, STATS = 10"  
Password:  
10 percent processed.  
20 percent processed.  
30 percent processed.  
40 percent processed.  
50 percent processed.  
60 percent processed.  
70 percent processed.  
80 percent processed.  
90 percent processed.  
Processed 24312 pages for database 'AdventureWorks', file 'AdventureWorks2014_Data' on file 1.  
100 percent processed.  
Processed 5 pages for database 'AdventureWorks', file 'AdventureWorks2014_Log' on file 1.  
BACKUP DATABASE successfully processed 24317 pages in 7.860 seconds (24.169 MB/sec).  
root@SQLServervNext-2:/home/dbadmin#
```

SQL Server vNext im Container

```
dbadmin@docker:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
78445dd45222: Pull complete
Digest: sha256:c5515758d4c5e1e838e9cd307f6c6a0d620b5e07e6f927b07d05f6d12a1ac8d7
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://cloud.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/engine/userguide/

dbadmin@docker:~$
```

SQL Server vNext im Container

```
dbadmin@docker:~$ sudo docker pull microsoft/mssql-server-linux
Using default tag: latest
latest: Pulling from microsoft/mssql-server-linux
aed15891ba52: Pull complete
773ae8583d14: Pull complete
d1d48771f782: Pull complete
cd3d6cd6c0cf: Pull complete
8ff6f8a9120c: Pull complete
1fd7e8b10447: Pull complete
bd485157db89: Pull complete
273a1970ce9c: Pull complete
6c4efc30bdcd: Pull complete
fa79d909c949: Pull complete
3251c885d451: Pull complete
Digest: sha256:c0588fe22b96140f73b27425ddc60efeae9a4fa9701cf3b22cccdab15397486
Status: Downloaded newer image for microsoft/mssql-server-linux:latest
dbadmin@docker:~$
```

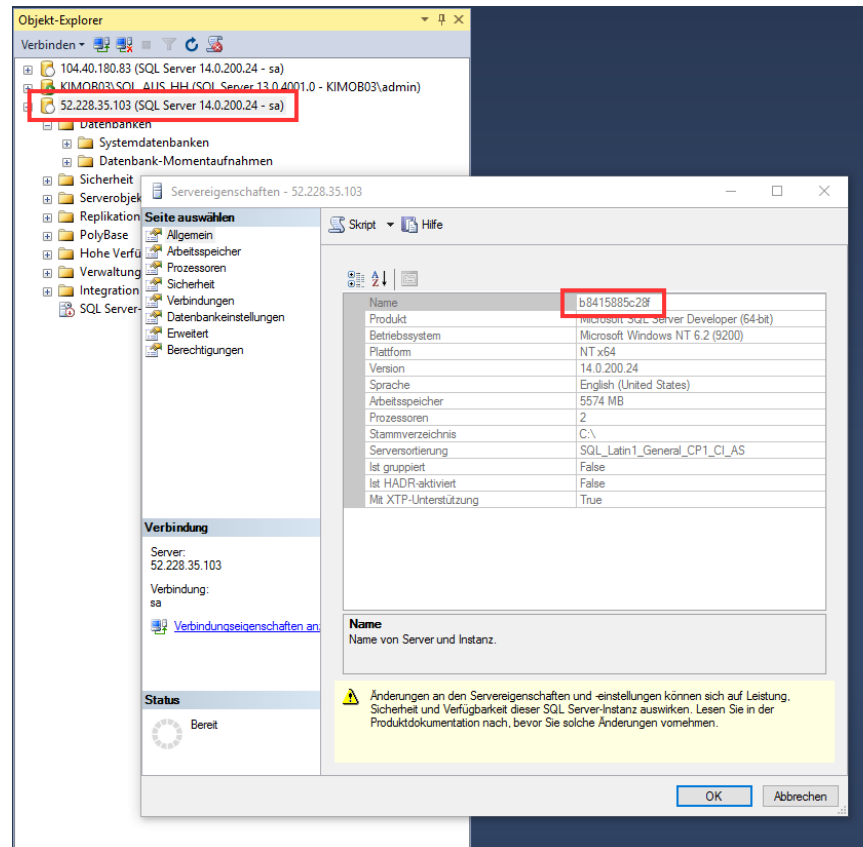

SQL Server vNext im Container

```
sudo docker run -e 'ACCEPT_EULA=Y' -e 'SA_PASSWORD=<YourStrong!Passw0rd>' -p 1433:1433  
-d microsoft/mssql-server-linux
```

```
dbadmin@docker:~$ docker run -e 'ACCEPT_EULA=Y' -e 'SA_PASSWORD=  
docker: Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Post http://%2Fvar%2Frun%2Fd  
docker.sock: connect: permission denied.  
See 'docker run --help'.  
dbadmin@docker:~$ sudo docker run -e 'ACCEPT_EULA=Y' -e 'SA_PASSWORD=  
b8415885c28f825ab2724300e6a70425204f8817b6dc922ed2de2ecd2dbc265b  
dbadmin@docker:~$
```

```
sudo docker run -e 'ACCEPT_EULA=Y' -e 'SA_PASSWORD=<YourStrong!Passw0rd>' -p 1433:1433  
-v <host directory>:/var/opt/mssql -d microsoft/mssql-server-linux
```

SQL Server vNext im Container



Wege zum Ziel

Logging Shipping => SQL Server vNext

<https://dbafromthecold.wordpress.com/2017/01/25/transaction-log-shipping-in-sql-server-on-linux/>

High Availability / Security mit HADR
oder Availability Groups

<https://blogs.technet.microsoft.com/dataplatforminsider/2016/11/30/sql-server-on-linux-high-availability-and-security/>



Offene Fragen ???



Infrastruktur auf Azure IaaS und SQL Server Best Practices

Patrick Heyde

- In Vorbereitung
 - Für dieses PASS Essential kann man sich bei
 - *E-Mail:* registrierung@sqlpass.de
 - **als Interessent unverbindlich vormerken lassen.** Wenn Ort und Zeitpunkt feststehen, werden alle Interessenten per Mail informiert und können dann entscheiden, ob man sich anmelden möchte.

Infrastruktur auf Azure IaaS und SQL Server Best Practices

- Neue IT-Regeln - Azure Infrastruktur Regeln für den SQL Server
 - Ressource Gruppen, Storage Account, Virtuelle Netzwerke, Virtuelle Maschinen
- Azure Storage für den SQL Server
 - Premium Storage, Storage Pools, BlockBlob, PageBlob
- SQL Server Konfigurationseinstellungen für Azure Virtuelle Maschinen
 - TEMPDB – Positionierung und die Auswirkungen auf die Performance
- Azure Management Portal & Azure Account Portal
- Dynamische Hardware – IaaS und deren Anwendung und Nutzen
- Azure Kosten bestimmen und vorab Planen
- An konkreten Beispielen, die mit dem eigenen Laptop und der eigenen Azure Subscription, wird der SQL Server über das Azure Management Portal aufgebaut.

Professionelles Hosting in Azure mit SQL Server on Azure

Patrick Heyde

- In Vorbereitung
 - Für dieses PASS Essential kann man sich bei
 - *E-Mail:* registrierung@sqlpass.de
 - **als Interessent unverbindlich vormerken lassen.** Wenn Ort und Zeitpunkt feststehen, werden alle Interessenten per Mail informiert und können dann entscheiden, ob man sich anmelden möchte.

Professionelles Hosting in Azure mit SQL Server on Azure

- Neue IT-Regeln - Azure Infrastruktur Regeln für den SQL Server
 - Ressource Gruppen, Storage Account, Virtuelle Netzwerke, Virtuelle Maschinen
- Aufbau von SQL Server in Azure Virtuelle Maschinen
- Automatisierung mit Azure Powershell cmdlets
 - Aufbau & Setup von SQL Server auf Azure Virtuelle Maschinen
 - Skalierung von SQL Server: Scale Up, Scale Down
 - Verwendung von Azure Custom Script Extentsion
 - Verwendung von Azure Powerhsell DSC
- Aufbau von Dynamischen SQL Servern in IaaS und deren Anwendung
- Azure Kosten bestimmen und vorab Planen

Patrick Heyde

- Patrick Heyde ist Technical Evangelist mit dem Schwerpunkt Microsoft Azure und zählte zu den ersten Nutzern der Azure IaaS Plattform-Features aus dem Jahr 2012.
- Mit seiner vorherigen 6-jährigen SharePoint- & SQL Server-Vergangenheit als Consultant und der Support Erfahrung als Support Escalation Engineer hat er klassische IT-Strukturen aufgebaut und betrieben.
- Zu seinem Spezialgebiet zählt die Migrations- & Architekturberatung von klassischen IT-Strukturen nach Azure IaaS & PaaS für den Aufbau und anschließenden Hosting Betrieb mit neuen Automatisierungsanteilen. Parallel zu den Projekten findet man ihn als Sprecher auf Konferenzen und Usergroups.
- Mehr über ihn findet man auf seinem [Technet-Blog](#) und auf Twitter @patrickheyde.

