SOLISYON

SQL Server 2019
Big Data Clusters

Ben Weissman
@bweissman

What we need

- All Materials on: https://github.com/bweissman/code/tree/master/SQLSatRheinlandPrecon

- Azure Subscription with Access to AKS
- Ideally: Virtual machine in Azure with 16 GB+ RAM and 4+ Cores
- SQL Server 2019 CTP 2.5 installer including ISO (Download Media)
  - https://www.microsoft.com/en-us/sql-server/sql-server-2019
- Kaggle.com Account
  - Flight Delay Dataset
    - https://www.kaggle.com/usdot/flight-delays

Who am I?

Ben Weissman

@bweissman

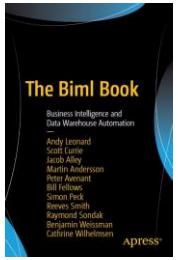
b.weissman@solisyon.de

http://biml-blog.de/

I ♥ DATA

Please Talk Data To Me

SOLISYON



The Biml Book

Business Intelligence and Data Warehouse Automation

Andy Leonard
Scott Currie
Jacob Alley
Martin Andersson
Peter Avenant
Bill Fellows
Simon Peck
Reeves Smith
Raymond Sondak
Benjamin Weissman
Cathrine Wilhelmsen

Apress®

BimlHero CERTIFIED EXPERT

MVP Microsoft® Most Valuable Professional

Microsoft CERTIFIED Solutions Associate

Machine Learning

Microsoft CERTIFIED Solutions Expert

Data Management and Analytics

MICROSOFT PROFESSIONAL PROGRAM

Data Science

Big Data

Artificial Intelligence

Data Analysis

FRIEND OF redgate 2019

GENESEE ACADEMY World-Class Training

Certified Data Vault Modeler

Der Data Platform Podcast mit Biml Ben, Mr. T und Angry Frank

Adaptive Query Processing ADF Azure Azure Data Studio **Azure Notebooks** Azure Stack Big Data Clusters **Biml** Black Panther Business Application Summit 2018 Data Platform **Data Platform Summit** dbatools **Docker** Flensburger Radler Alkoholfrei GDPR **Git Hub** Ignite Jupyter Notebooks Kubernetes **Las Vegas** Lissabon Microsoft Professional Program MPP **PASS Camp** PASS Deutschland e.V. PASS Essentials PASS Summit **Power BI** PowerShell Query Folding Regionalgruppen Solo SQL Management Studio 18 - Preview SQL Operations Studio **SQL Saturday** SQL Server 2019 Tabular Tomb Raider **tSQLt** TugaIT Visual Studio Code **WDC**

Ben Weissman
Biml Ben

Tillmann Eitelberg
Mr. T

Frank Geisler
Angry Frank

11
Episoden

2396
Downloads

1147
Sendeminuten

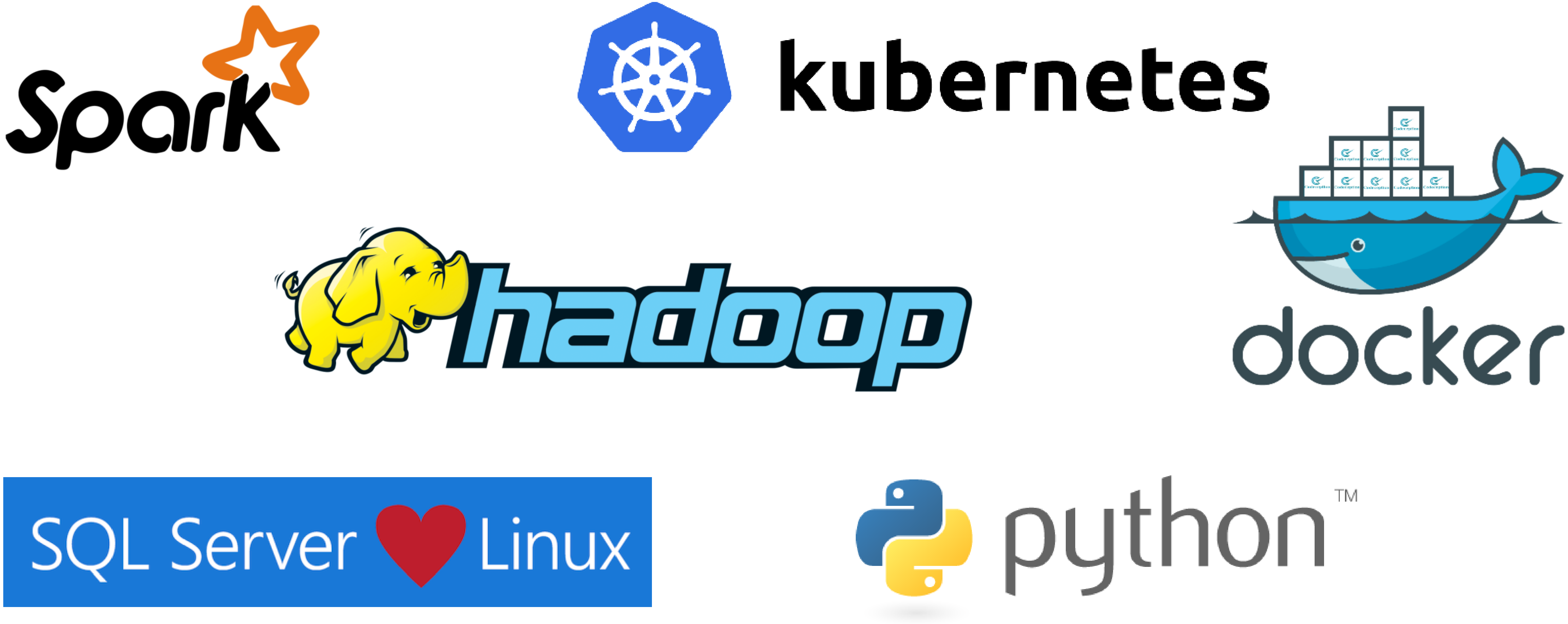
12
Gäste

https://www.pleasetalkdatatome.de

Fun Facts on Big Data Clusters

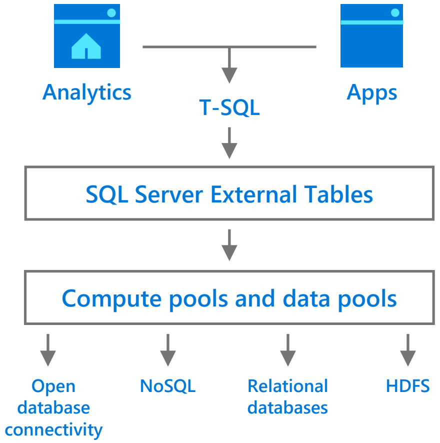
- Some parts only run on Linux
- It‘s a „box product first“ feature set
- It‘s actually not ONE feature but a huge feature set
- It‘s name is a bit misleading – not all of it is a cluster
- Some parts are currently in semi-private preview

What to look at before getting started…



So what is a Big Data Cluster in SQL 2019?!

Data virtualization



Analytics

Apps

T-SQL

SQL Server External Tables

Compute pools and data pools

Open database connectivity

NoSQL

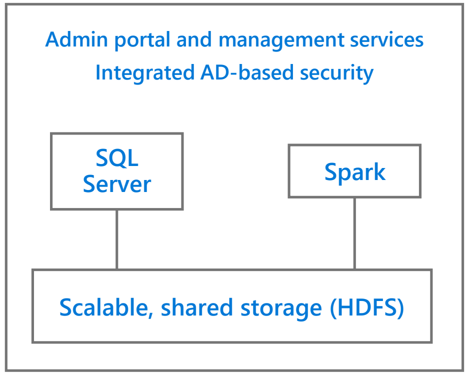
Relational databases

HDFS

Combine data from many sources without moving or replicating it

Scale out compute and caching to boost performance

Managed SQL Server, Spark, and data lake



Admin portal and management services

Integrated AD-based security

SQL Server

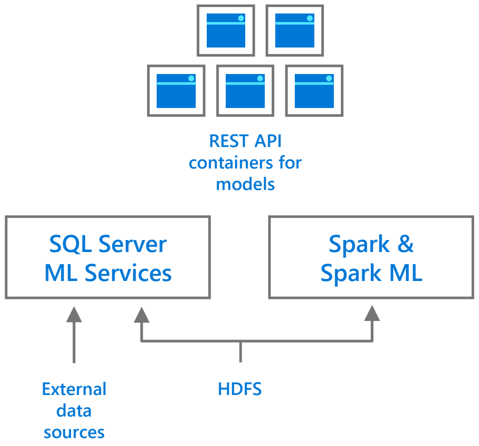
Spark

Scalable, shared storage (HDFS)

Store high volume data in a data lake and access it easily using either SQL or Spark

Management services, admin portal, and integrated security make it all easy to manage

Complete AI platform



REST API containers for models

SQL Server ML Services

Spark & Spark ML

External data sources

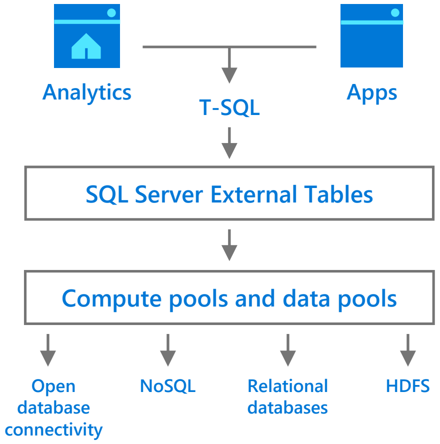
HDFS

Easily feed integrated data from many sources to your model training

Ingest and prep data and then train, store, and operationalize your models all in one system

Data Virtualization – Is this the END of SSIS?!

Data virtualization


Analytics
T-SQL
Apps
SQL Server External Tables
Compute pools and data pools
Open database connectivity
NoSQL
Relational databases
HDFS

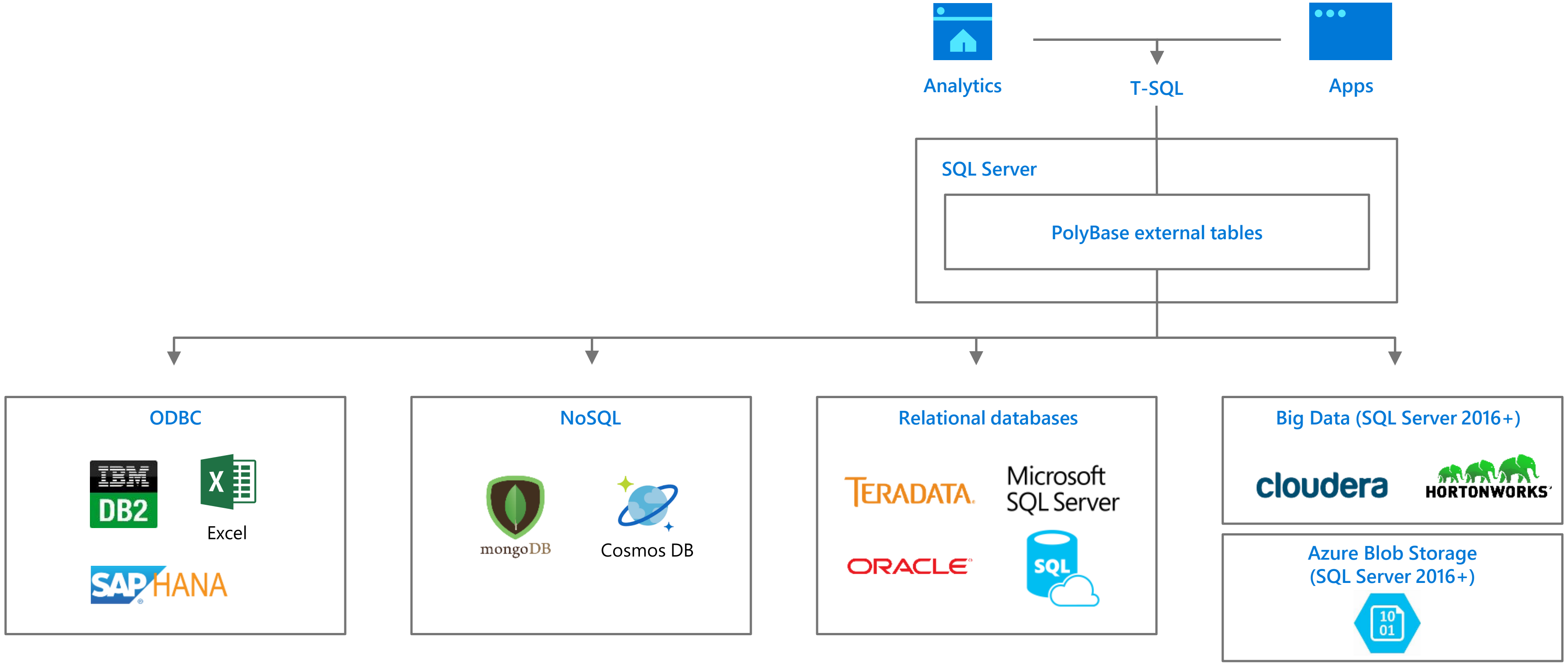
Combine data from many sources without moving or replicating it

Scale out compute and caching to boost performance

This slide: © by Microsoft

Polybase in SQL Server 2019

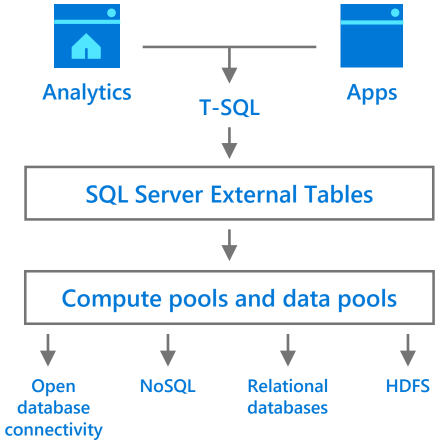
Easily combine across relational and non-relational data stores



This slide: © by Microsoft

Data Virtualization – So it‘s a linked server?

Data virtualization


Analytics
T-SQL
Apps
SQL Server External Tables
Compute pools and data pools
Open database connectivity
NoSQL
Relational databases
HDFS

Combine data from many sources without moving or replicating it

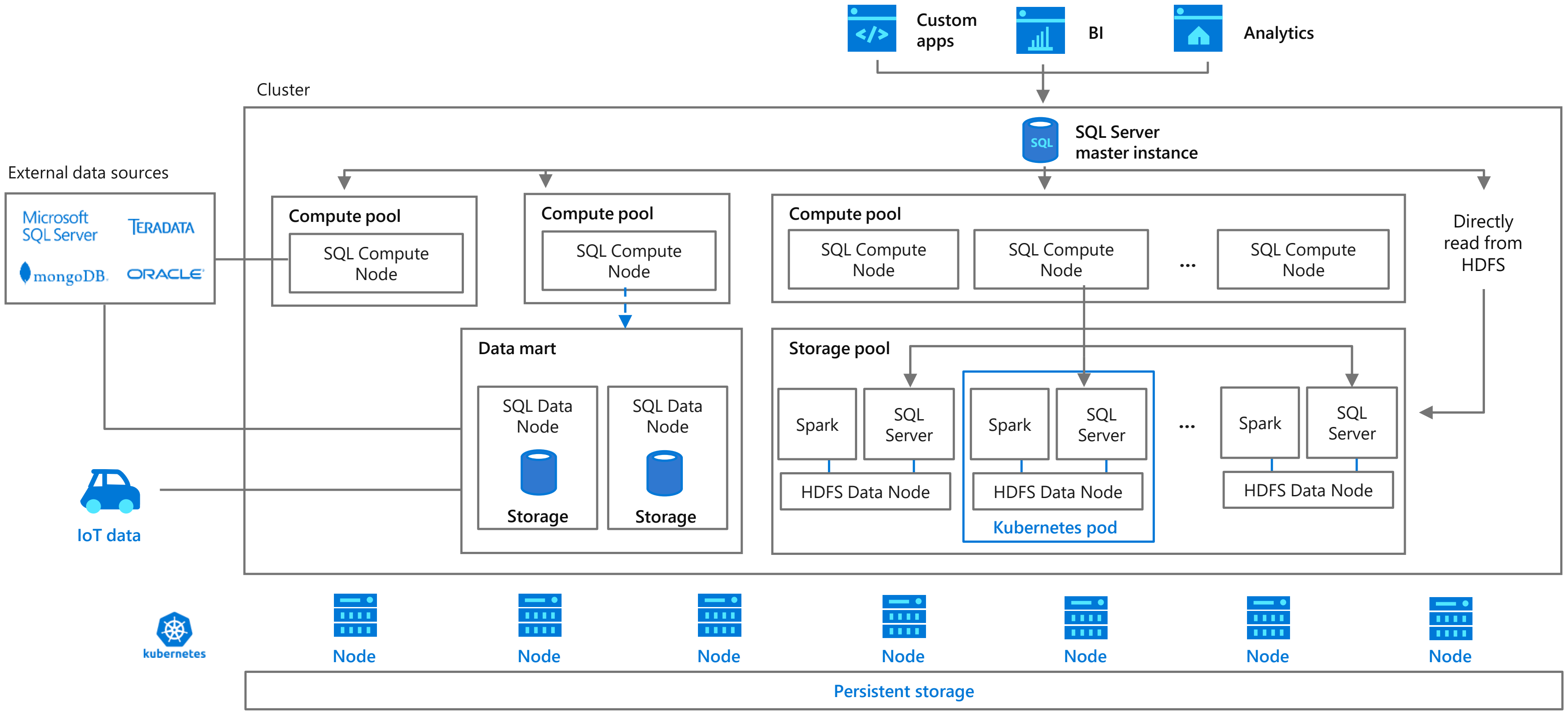
Scale out compute and caching to boost performance

Linked Servers

PolyBase External tables

This slide: © by Microsoft

So why do they call it a cluster?!

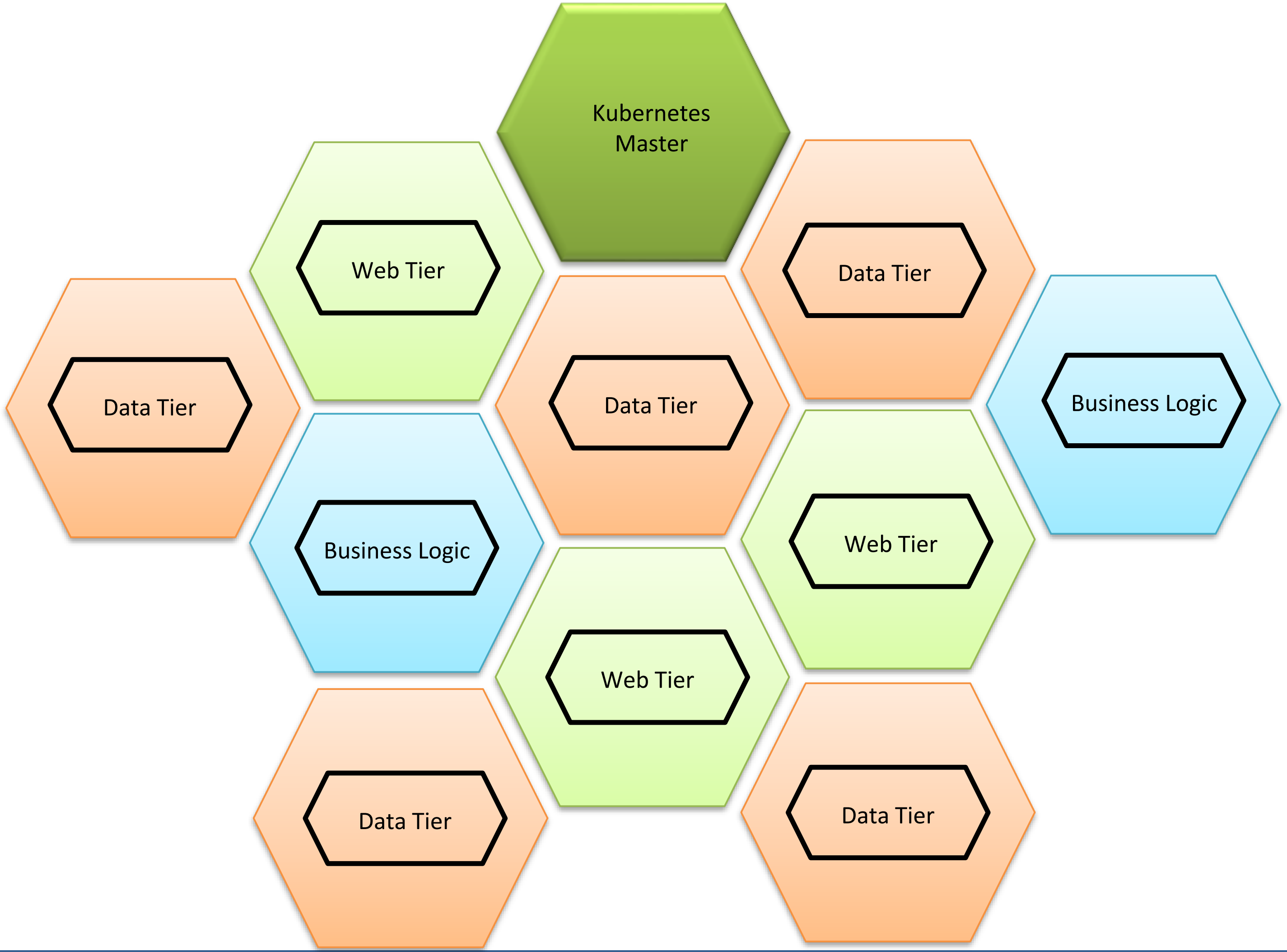


This slide: © by Microsoft

Generic Kubernetes Cluster

Scale by Purpose

kubernetes



How can I get it installed? – Polybase only

- Install Java JRE
- Get the latest CTP from http://microsoft.com/sql
- Install SQL Server on Windows or Linux including Polybase
- Use EVALUTATION edition!
- Enable Polybase after installation:

```
exec sp_configure @configname = 'polybase enabled', @configvalue = 1;
RECONFIGURE
```

- Restart SQL Server
- Install Azure Data Studio
- Install the vNext Extension for Azure Data Studio

How can I get it installed? – The full package

- Sign up for the preview program: https://aka.ms/eapsignup
- Install Kubernetes-CLI, MSSQLCTL, Python, azure-cli, curl*
- Install Azure Data Studio
  - Add vNext Extension
- Decide on a Kubernetes environment
  - Docker or Minikube
  - AKS
  - Something completely different ☺
  - (many but not all are supported)
- Set environment variables**
- Deploy the cluster using

```
mssqlctl create cluster <your-cluster-name>
```

- When using AKS, consider this script:

  https://github.com/Microsoft/sql-server-samples/tree/master/samples/features/sql-big-data-cluster/deployment

NETGEAR ProSAFE Gigabit Switch GS108
NETGEAR ProSAFE Gigabit Switch GS105
NETGEAR ProSAFE Gigabit Switch GS108

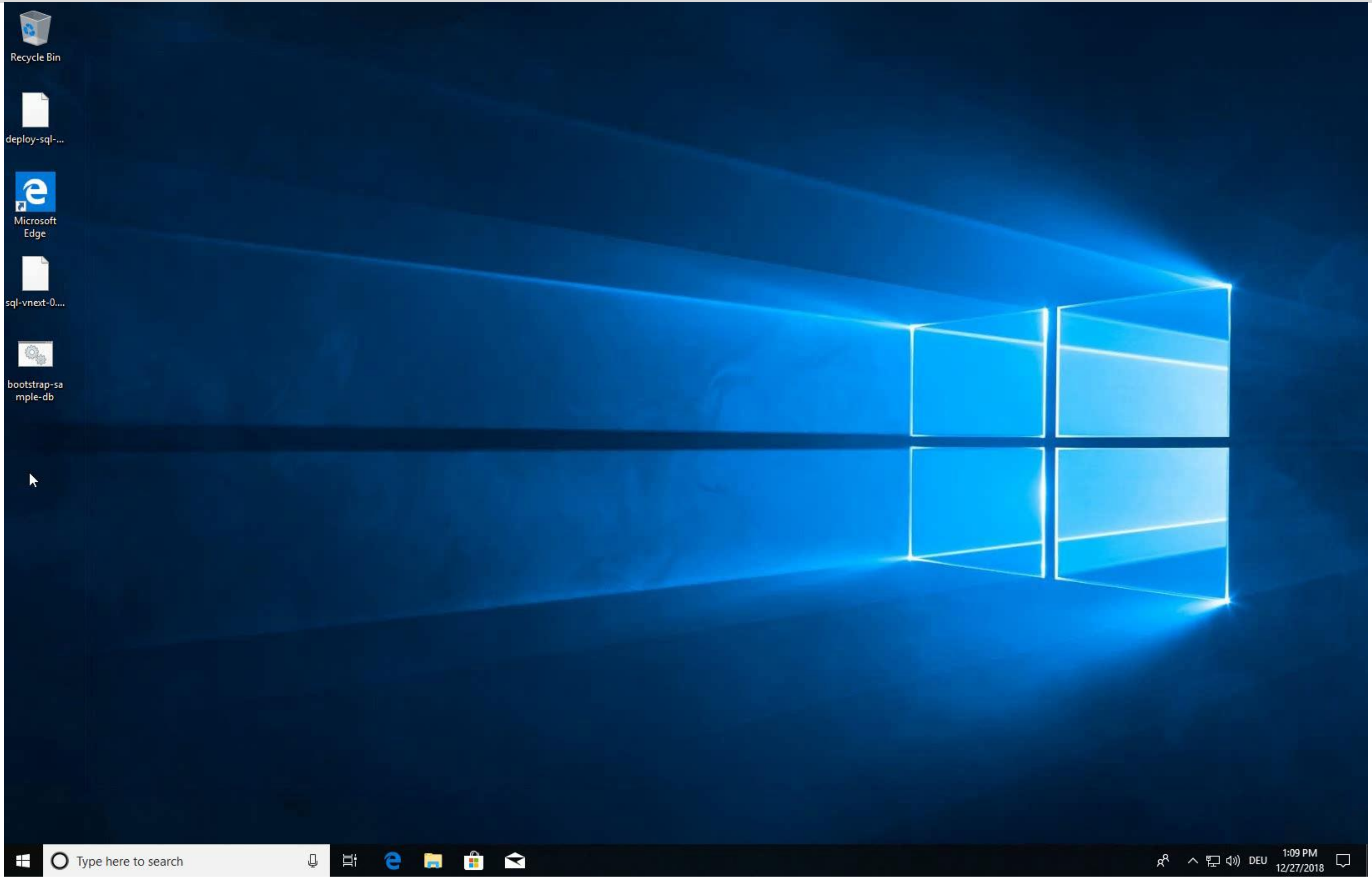
Picture: © Klaus Aschenbrenner

It‘s Demo Time: Kubernetes Cluster

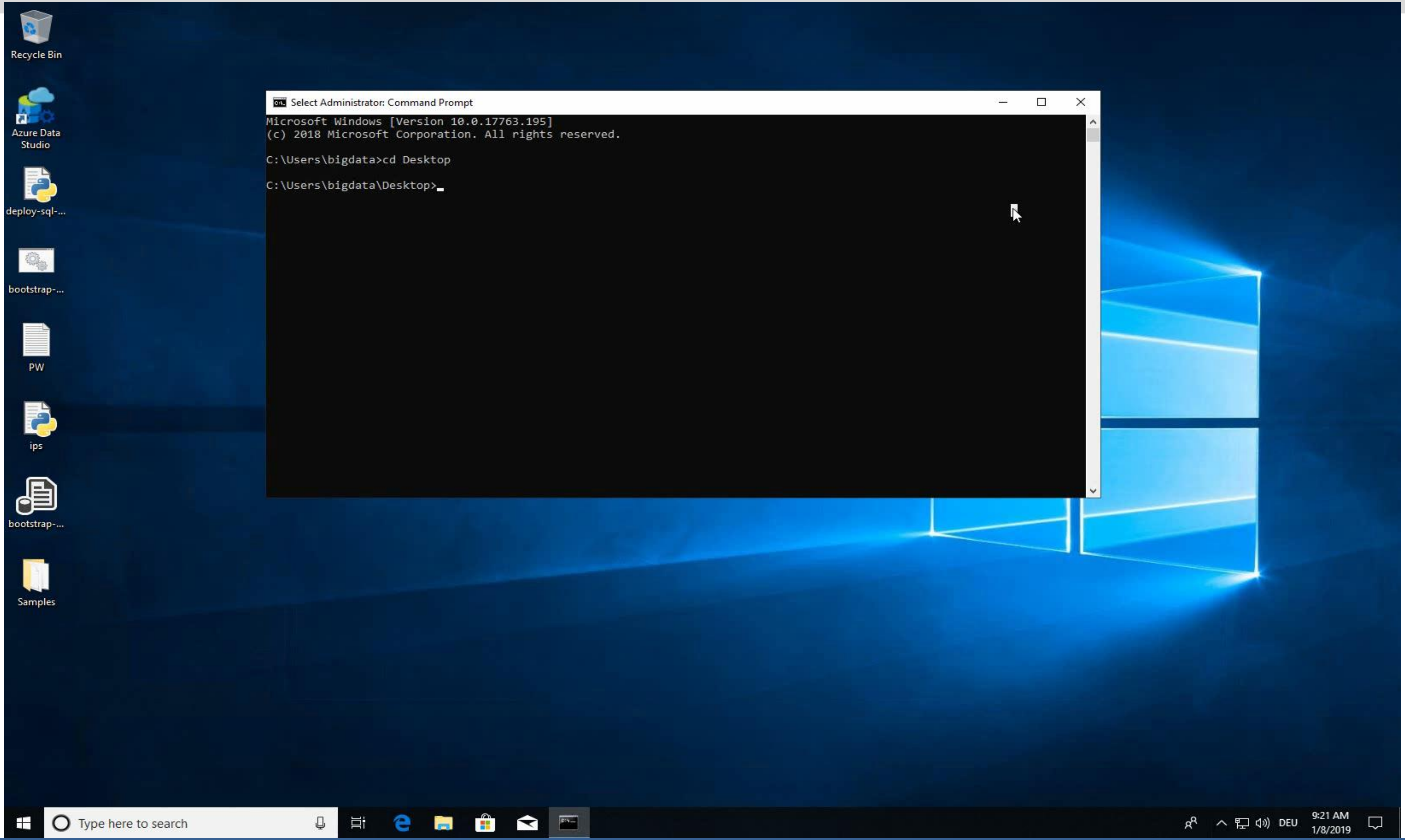
- Deploying a cluster with some sample data* (yay! videos ☺)
- The Cluster Portal
- Play with it using T-SQL
  - Query HDFS Data
  - Write/Read Data from Data Pool
- Play with it using Notebooks
  - Read/Analyze Date with Spark
  - Train and query a ML Model

*https://github.com/Microsoft/sql-server-samples/tree/master/samples/features/sql-big-data-cluster/

Kubernetes Cluster – Install Prerequisits



Kubernetes Cluster – Install Cluster (incl. AKS)

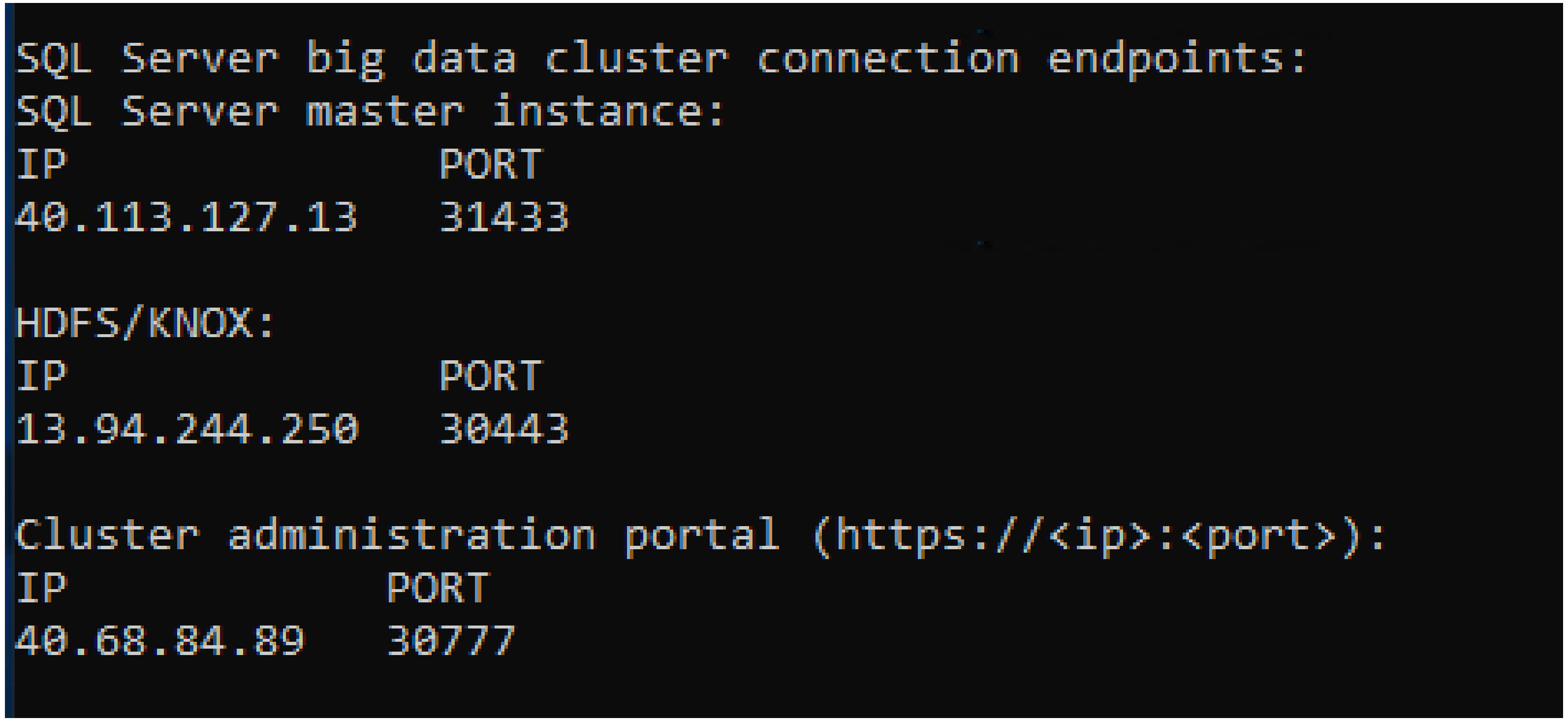


```
Microsoft Windows [Version 10.0.17763.195]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\bigdata>cd Desktop

C:\Users\bigdata\Desktop>_
```

Kubernetes Cluster – Install Cluster (incl. AKS)



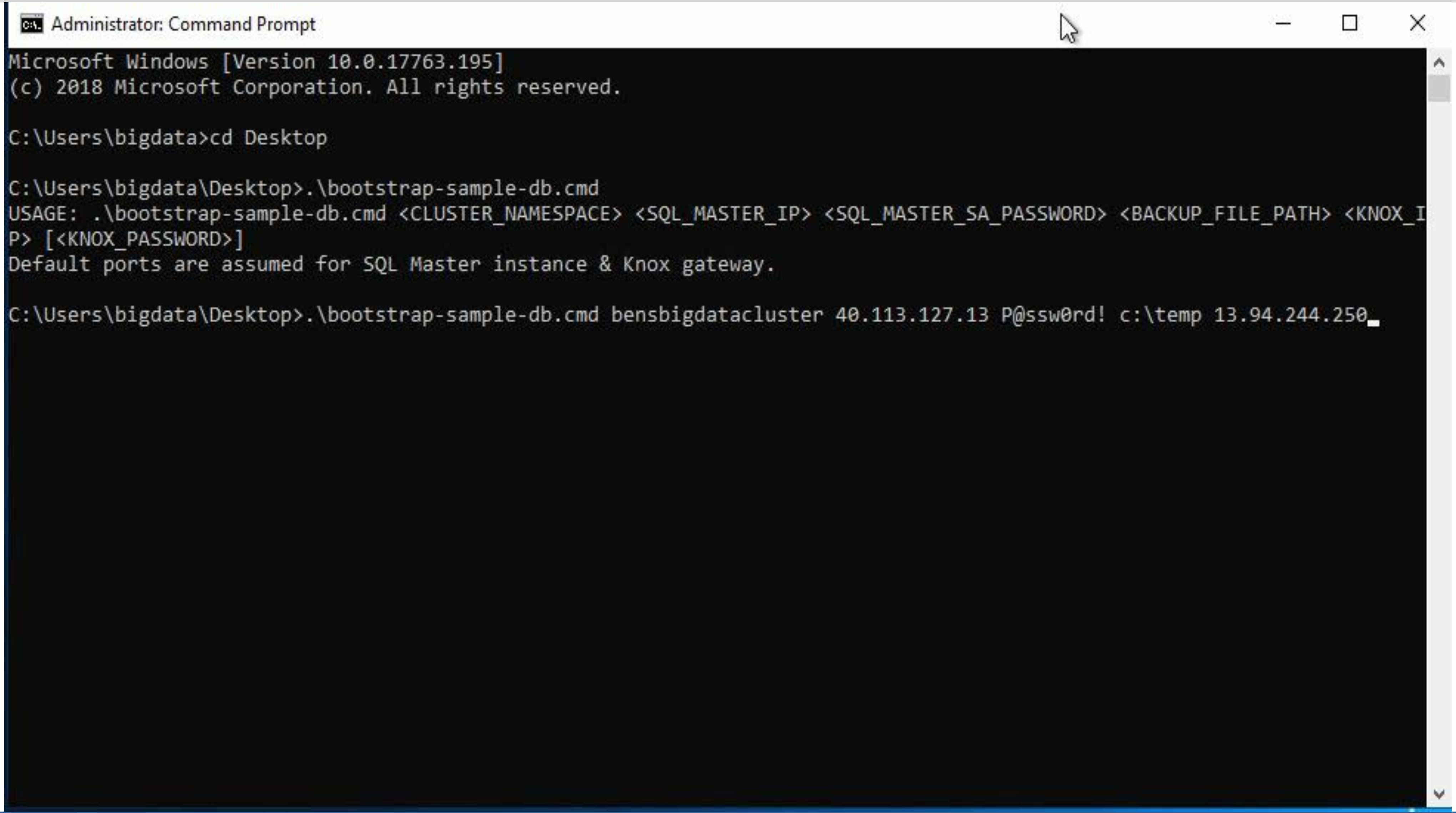
```
SQL Server big data cluster connection endpoints:
SQL Server master instance:
IP              PORT
40.113.127.13   31433

HDFS/KNOX:
IP              PORT
13.94.244.250   30443

Cluster administration portal (https://<ip>:<port>):
IP            PORT
40.68.84.89   30777
```

*If you forget about these…
kubectl get service -n <clustername>

Kubernetes Cluster – Install Sample Data



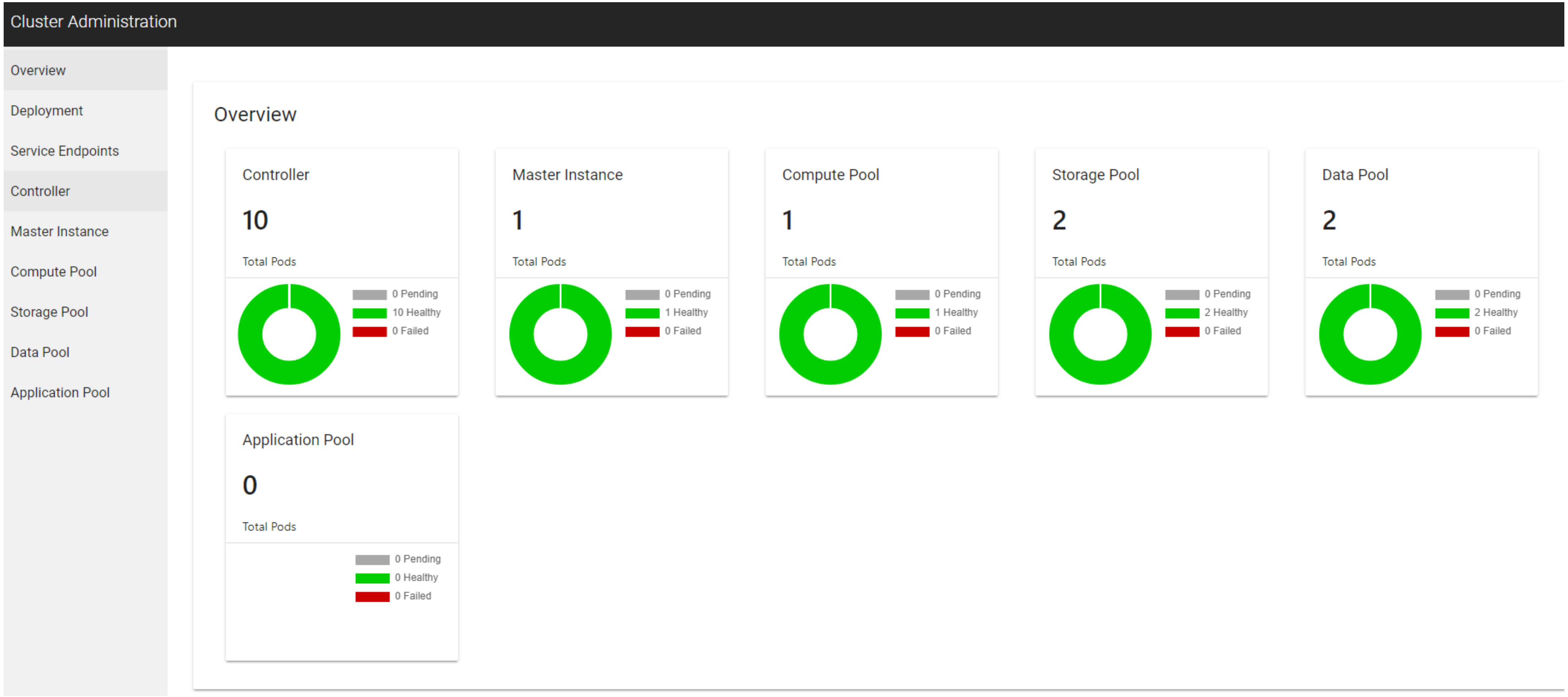
```
Microsoft Windows [Version 10.0.17763.195]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\bigdata>cd Desktop

C:\Users\bigdata\Desktop>.\bootstrap-sample-db.cmd
USAGE: .\bootstrap-sample-db.cmd <CLUSTER_NAMESPACE> <SQL_MASTER_IP> <SQL_MASTER_SA_PASSWORD> <BACKUP_FILE_PATH> <KNOX_I
P> [<KNOX_PASSWORD>]
Default ports are assumed for SQL Master instance & Knox gateway.

C:\Users\bigdata\Desktop>.\bootstrap-sample-db.cmd bensbigdatacluster 40.113.127.13 P@ssw0rd! c:\temp 13.94.244.250
```

Kubernetes Cluster – Portal



Cluster Administration

- Overview
- Deployment
- Service Endpoints
- Controller
- Master Instance
- Compute Pool
- Storage Pool
- Data Pool
- Application Pool

Overview

Controller

10

Total Pods

⬜ 0 Pending
🟩 10 Healthy
🟥 0 Failed

Master Instance

1

Total Pods

⬜ 0 Pending
🟩 1 Healthy
🟥 0 Failed

Compute Pool

1

Total Pods

⬜ 0 Pending
🟩 1 Healthy
🟥 0 Failed

Storage Pool

2

Total Pods

⬜ 0 Pending
🟩 2 Healthy
🟥 0 Failed

Data Pool

2

Total Pods

⬜ 0 Pending
🟩 2 Healthy
🟥 0 Failed

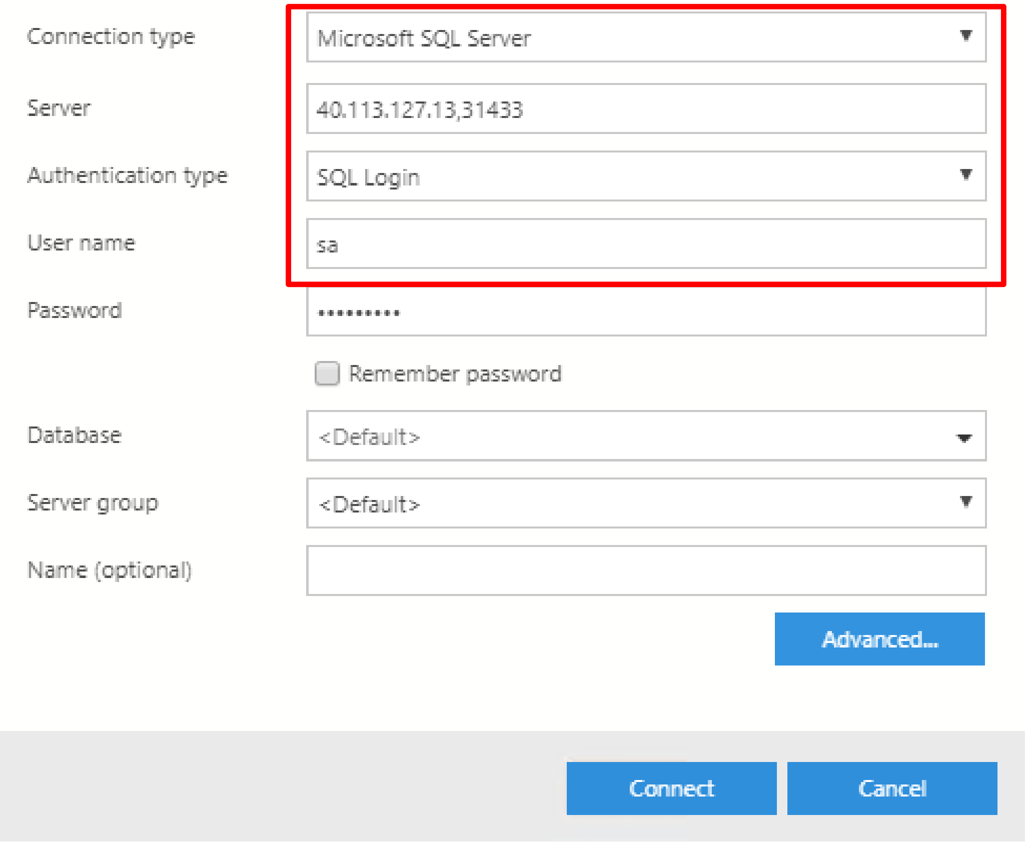
Application Pool

0

Total Pods

⬜ 0 Pending
🟩 0 Healthy
🟥 0 Failed

Kubernetes Cluster – Connect in ADS



Connection type: Microsoft SQL Server

Server: 40.113.127.13,31433

Authentication type: SQL Login

User name: sa

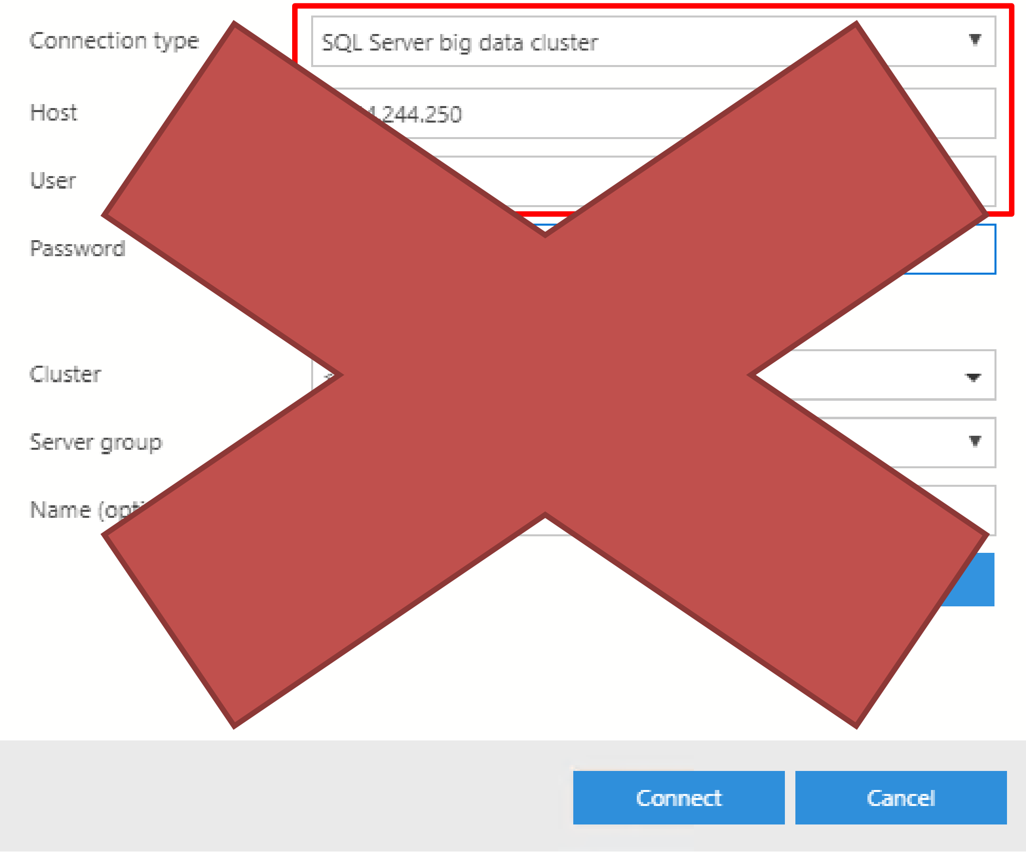
Password: ••••••••••

☐ Remember password

Database: <Default>

Server group: <Default>

Name (optional):



Connection type: SQL Server big data cluster

Host: .244.250

User

Password

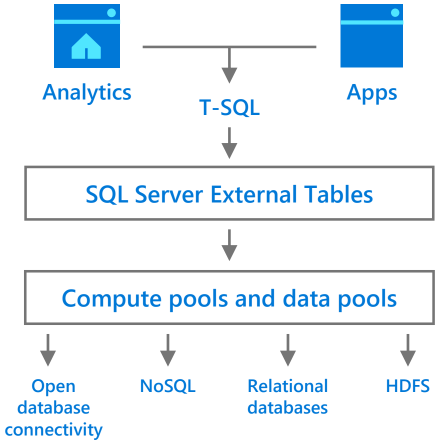
Cluster

Server group

Name (opt

So what is a Big Data Cluster in SQL 2019?!

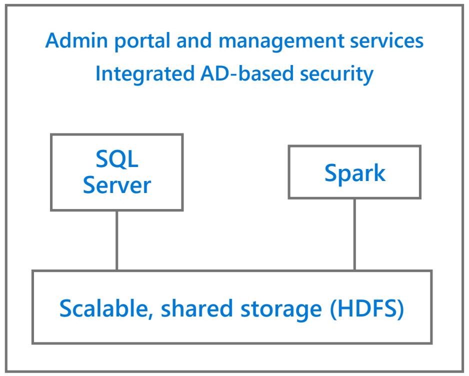
Data virtualization


Analytics
T-SQL
Apps
SQL Server External Tables
Compute pools and data pools
Open database connectivity
NoSQL
Relational databases
HDFS

- No Data redundancy
- Real time data

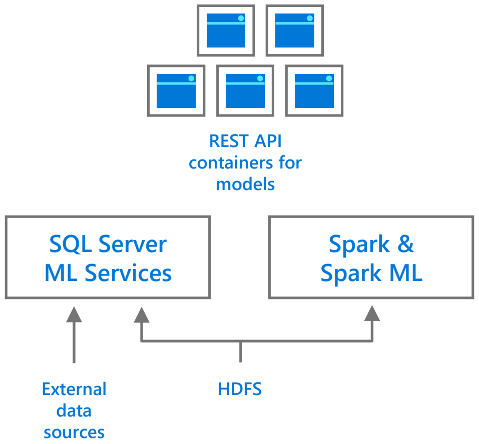
- No extra indexing
- Extra load on source
- Read only

Managed SQL Server, Spark, and data lake


Admin portal and management services
Integrated AD-based security
SQL Server
Spark
Scalable, shared storage (HDFS)

- Store high volume data in a data lake and access it easily using either SQL or Spark
- Management services, admin portal, and integrated security make it all easy to manage

Complete AI platform



- Easily feed integrated data from many sources to your model training
- Ingest and prep data and then train, store, and operationalize your models all in one system

This slide: © by Microsoft

Let‘s go! – Install Chocolatey

Run in Powershell:

```
Set-ExecutionPolicy Bypass -Scope Process -Force; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))
```

Agenda - Standalone PolyBase Installation

- Prerequistes
- Install SQL Server
- Enable Polybase
- Install Azure Data Studio
- Add Extension
- Add external tables from Azure SQL DB
    - Automate with Biml:
        - https://www.solisyon.de/biml-polybase-external-tables/
- Query those tables

Prerequisites for Polybase and client tools

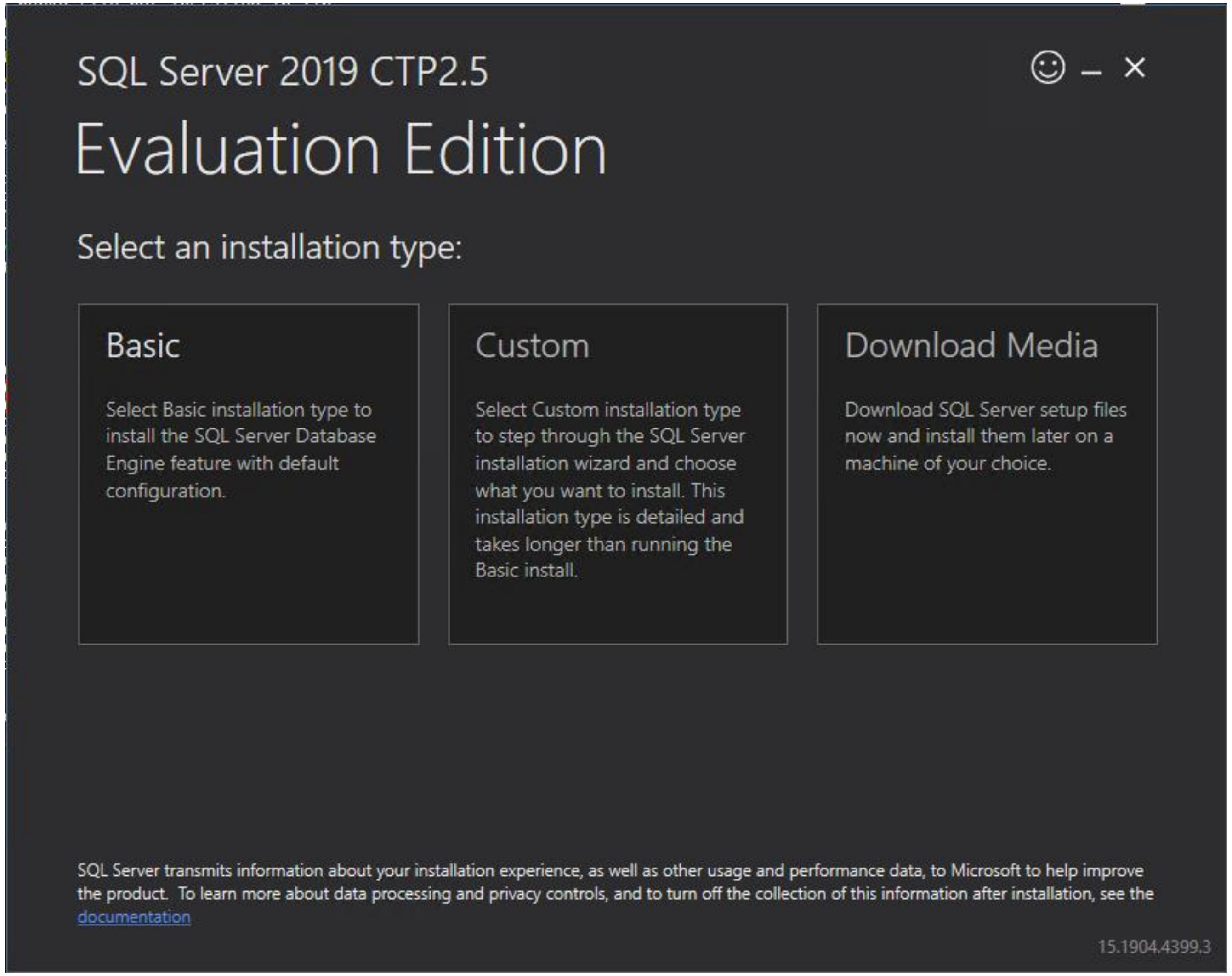
```
choco install jre8 -y
choco install vcredist2012 -y
choco install azure-data-studio -y
choco install sql-server-management-studio -y
```

Download ADS Extension from:

https://docs.microsoft.com/en-us/sql/azure-data-studio/sql-server-2019-extension?view=sqlallproducts-allversions

Install extension to Azure Data Studio

Install SQL Server



SQL Server 2019 CTP2.5

Evaluation Edition

Select an installation type:

Basic

Select Basic installation type to install the SQL Server Database Engine feature with default configuration.

Custom

Select Custom installation type to step through the SQL Server installation wizard and choose what you want to install. This installation type is detailed and takes longer than running the Basic install.

Download Media

Download SQL Server setup files now and install them later on a machine of your choice.

SQL Server transmits information about your installation experience, as well as other usage and performance data, to Microsoft to help improve the product. To learn more about data processing and privacy controls, and to turn off the collection of this information after installation, see the documentation

15.1904.4399.3

Enable PolyBase

```
exec sp_configure @configname = 'polybase enabled', @configvalue = 1;
RECONFIGURE
```

Create Database (empty) BigDataFun

Restart SQL Server

Create external tables using ADS

Agenda – Full Cluster in AKS

- Prerequistes
- Deploy Cluster
- Add Database to Cluster
- Add CSVs to Cluster
- Add MS Sample Data to Cluster
- Query the Cluster

SOLISYON

Additional Prerequisites for AKS

```
choco install python3 -y
choco install sqlserver-cmdlineutils -y
$env:Path = [System.Environment]::GetEnvironmentVariable("Path","Machine") + ";" + [System.Environment]::GetEnvironmentVariable("Path","User")
python -m pip install --upgrade pip
python -m pip install requests
python -m pip install requests --upgrade
choco install curl -y
choco install kubernetes-cli -y
choco install notepadplusplus -y
choco install 7zip -y
pip3 install kubernetes
pip3 install -r https://private-repo.microsoft.com/python/ctp-2.5/mssqlctl/requirements.txt
choco install azure-cli -y
```

Deploy the Cluster

```
az login
```

```
curl -o deploy-sql-big-data-aks.py "https://raw.githubusercontent.com/Microsoft/sql-server-samples/master/samples/features/sql-big-data-cluster/deployment/aks/deploy-sql-big-data-aks.py"
python deploy-sql-big-data-aks.py
```

Subscription ID: From az login
RG Name: sqlsatprecon
Docker Username/PW: From Credentials
Azure Region: westeurope
VM Size: Default (if available)
Number of nodes: 1
Clustername: sqlsatprecon
PW: Default (MySQLBigData2019)
User: Default (admin)

WAIT ☺

Look at the cluster in the Azure Portal and Management Portal

Add Data – Adventureworks2014

```
curl -L -G "https://github.com/Microsoft/sql-server-samples/releases/download/adventureworks/AdventureWorks2014.bak" -o AdventureWorks2014.bak
```

```
kubectl cp AdventureWorks2014.bak sqlsatprecon/master-0:var/opt/mssql/data -c mssql-server
```

Connect to the Cluster Master Instance in Azure Data Studio

```
USE [master]
RESTORE DATABASE [AdventureWorks2014] FROM  DISK = N'/var/opt/mssql/data/AdventureWorks2014.bak' WITH  FILE = 1,  MOVE
N'AdventureWorks2014_Data' TO N'/var/opt/mssql/data/AdventureWorks2014_Data.mdf',  MOVE N'AdventureWorks2014_Log' TO
N'/var/opt/mssql/data/AdventureWorks2014_Log.ldf',  NOUNLOAD,  STATS = 5
```

Add Data – Flight Delays

→ Get it from Kaggle first!

Create Directory:

```
curl -i -L -k -u root:%KNOX_PASSWORD% -X PUT "https://%KNOX_ENDPOINT%/gateway/default/webhdfs/v1/FlightDelays?op=MKDIRS"
```

Add files in Azure Data Studio

Add Data – MS Sample Data

```
curl -o bootstrap-sample-db.cmd "https://raw.githubusercontent.com/Microsoft/sql-server-samples/master/samples/features/sql-big-data-cluster/bootstrap-sample-db.cmd"

curl -o bootstrap-sample-db.sql "https://raw.githubusercontent.com/Microsoft/sql-server-samples/master/samples/features/sql-big-data-cluster/bootstrap-sample-db.sql"

.\bootstrap-sample-db.cmd <CLUSTER_NAMESPACE> <SQL_MASTER_IP> <SQL_MASTER_SA_PASSWORD> <KNOX_IP> <KNOX_PASSWORD> --install-extra-samples

.\bootstrap-sample-db.cmd sqlsatprecon <SQL_MASTER_IP> MySQLBigData2019 <KNOX_IP> MySQLBigData2019 --install-extra-samples
```

Query the cluster – Flight Delays

```
SELECT TOP 10 *
FROM flights fl
   INNER JOIN airlines al
    ON fl.AIRLINE = al.IATA_CODE
    INNER JOIN airports ap
    ON fl.DESTINATION_AIRPORT = ap.IATA_CODE;
```

Query the cluster – MS Samples

https://github.com/microsoft/sql-server-samples/blob/master/samples/features/sql-big-data-cluster/spark/data-loading/transform-csv-files.ipynb

Easier:

```
curl -o transform.ipynb "https://raw.githubusercontent.com/microsoft/sql-server-samples/master/samples/features/sql-big-data-cluster/spark/data-loading/transform-csv-files.ipynb"
```

https://github.com/microsoft/sql-server-samples/blob/master/samples/features/sql-big-data-cluster/data-virtualization/storage-pool/web-clickstreams-hdfs-csv.sql

https://github.com/microsoft/sql-server-samples/blob/master/samples/features/sql-big-data-cluster/data-virtualization/storage-pool/product-reviews-hdfs-csv.sql

https://github.com/microsoft/sql-server-samples/blob/master/samples/features/sql-big-data-cluster/data-virtualization/storage-pool/web-clickstreams-hdfs-parquet.sql

https://github.com/microsoft/sql-server-samples/blob/master/samples/features/sql-big-data-cluster/data-pool/data-ingestion-sql.sql

```
curl -o 01-web-clickstreams-hdfs-csv.sql "https://raw.githubusercontent.com/microsoft/sql-server-samples/master/samples/features/sql-big-data-cluster/data-virtualization/storage-pool/web-clickstreams-hdfs-csv.sql"
curl -o 02-product-reviews-hdfs-csv.sql "https://raw.githubusercontent.com/microsoft/sql-server-samples/master/samples/features/sql-big-data-cluster/data-virtualization/storage-pool/product-reviews-hdfs-csv.sql"
curl -o 03-web-clickstreams-hdfs-parquet.sql "https://raw.githubusercontent.com/microsoft/sql-server-samples/master/samples/features/sql-big-data-cluster/data-virtualization/storage-pool/web-clickstreams-hdfs-parquet.sql"
curl -o 04-data-ingestion-sql.sql "https://raw.githubusercontent.com/microsoft/sql-server-samples/master/samples/features/sql-big-data-cluster/data-pool/data-ingestion-sql.sql"
```

Further Examples / Reading

https://github.com/microsoft/sqlworkshops/tree/master/sqlserver2019bigdataclusters

Any questions?

DON‘T FORGET TO DELETE YOUR AZURE RESOURCES!

Ben Weissman
@bweissman