A Lap Around SQL Server 2019 Big Data Cluster

Niels Berglund
niels.it.berglund@gmail.com
https://nielsberglund.com
@nielsberglund

PASS SQLSATURDAY CAPE TOWN | 14 SEP 2019

Thank Sponsors

Niels Obligatory Shameless Self Promo

- Software Architect - Derivco.
- Author - "First Look at SQL Server 2005 for Developers".
- Microsoft Data Platform MVP.
- Researcher / Instructor - DevelopMentor.
- Speaker - TechEd, DevWeek, SQL Pass, etc.
- Longtime user of SQL Server.
- Working closely with MS around SQL Server.

https://nielsberglund.com

Data Landscape

- We generate more and more data.
  - 2016 - 16.1 ZBs
  - 2025 - 163 ZBs
- The data is stored "all over the place".
- How do we manage all this data?

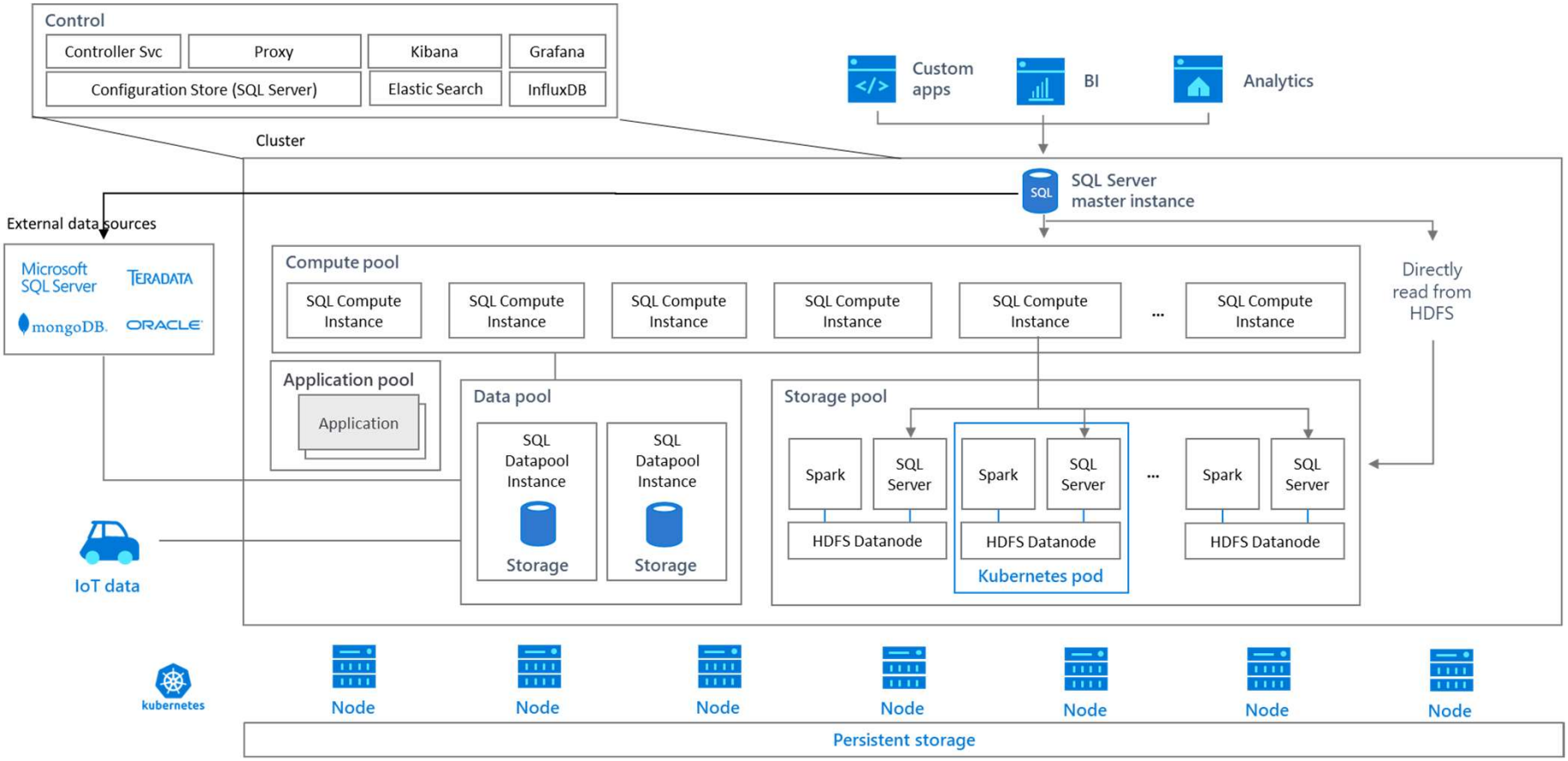
SQL Server - Intelligence Over All Your Data

- Manage all data
- Integrate all data
- Analyze all data

SQL Server 2019 Big Data Cluster

- Apache Spark, Hadoop HDFS "in the box".
- Extend SQL Server to store data in the teta-byte range.
- Store any kind of data.
- Linux containers on Kubernetes.

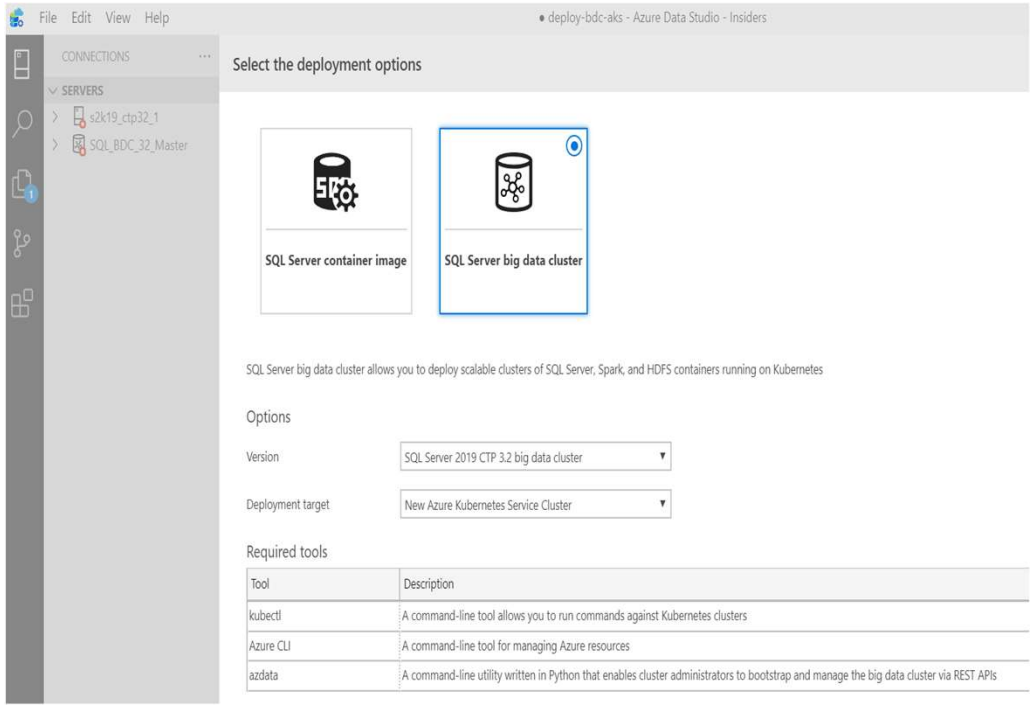
SQL Server 2019 Architecture on Kubernetes



https://nielsberglund.com

Deploying a BDC Cluster

- We are not in Kansas any more.
- Deployment via Python scripts.
  - Scripts for different environments.
- Deployment from Azure Data Studio deploy notebook.
  - Requires Azure Data Studio - Insiders build.
- Deploy to existing K8s cluster, or create new.
- During deployment set number of Nodes, etc.



https://nielsberglund.com

Managing a BDC Cluster

- Command line tools:
  - `kubectl`
  - `az` - Azure command line interface for managing Azure services.
  - `azdata` – Python command line tool for installing and managing BDC.

```
# login
az login
# set context
az aks get-credentials --name <aks_cluster_name>
                       --resource-group <azure_resource_group_name>
# get all pods
kubectl get pods --all-namespaces
# browse Kubernetes dashboard
az aks browse --resource-group <azure_resource_group_name>
              --name <aks_cluster_name>
# retrieve endpoints
azdata bdc endpoint list
```

Data Virtualization - PolyBase

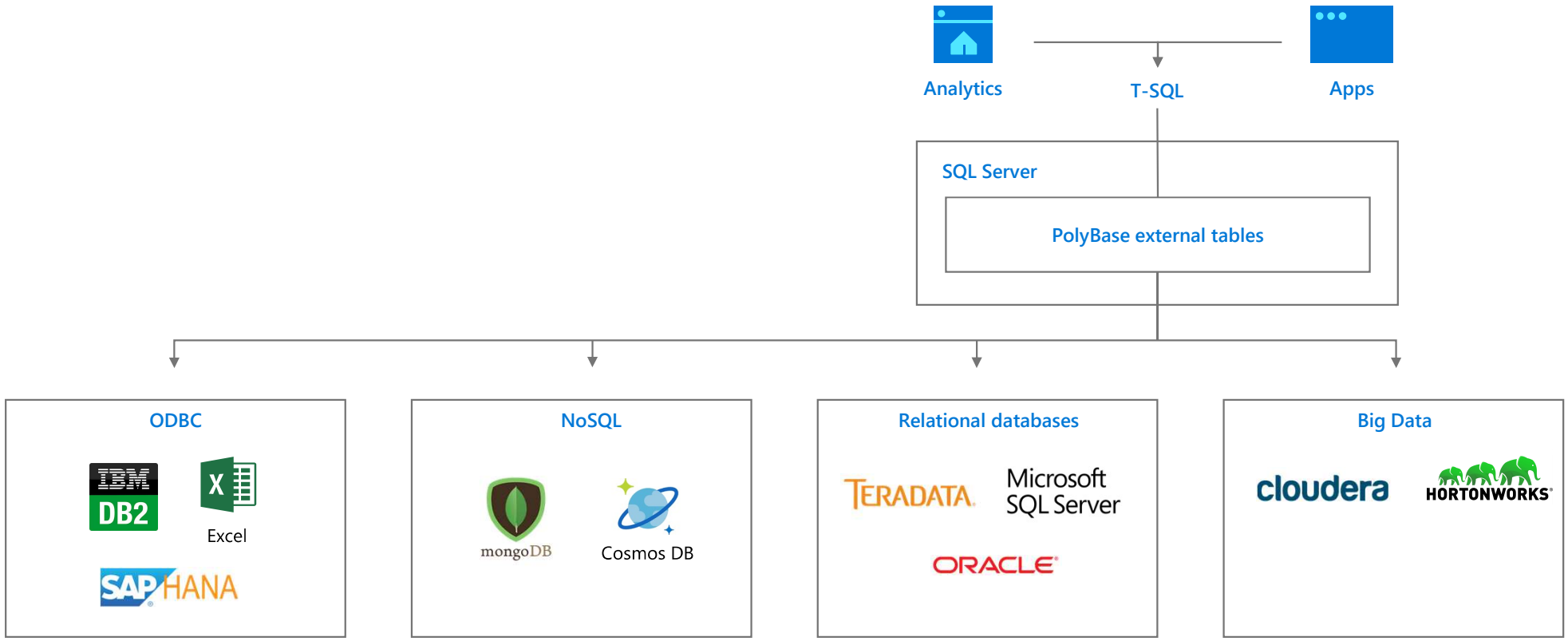
Linked Servers

- Instance scoped object
- Uses OLEDB providers
- Supports both read/write & pass-through statements
- Queries are single-threaded & push-down supported
- Separate configuration needed for each instance in Always On Availability Group

PolyBase External tables

- Database scoped object
- Uses ODBC drivers
- Supports read-only operations only. Will be expanded in future
- Queries can be scaled-out & push-down supported
- No separate configuration needed for Always On Availability Group

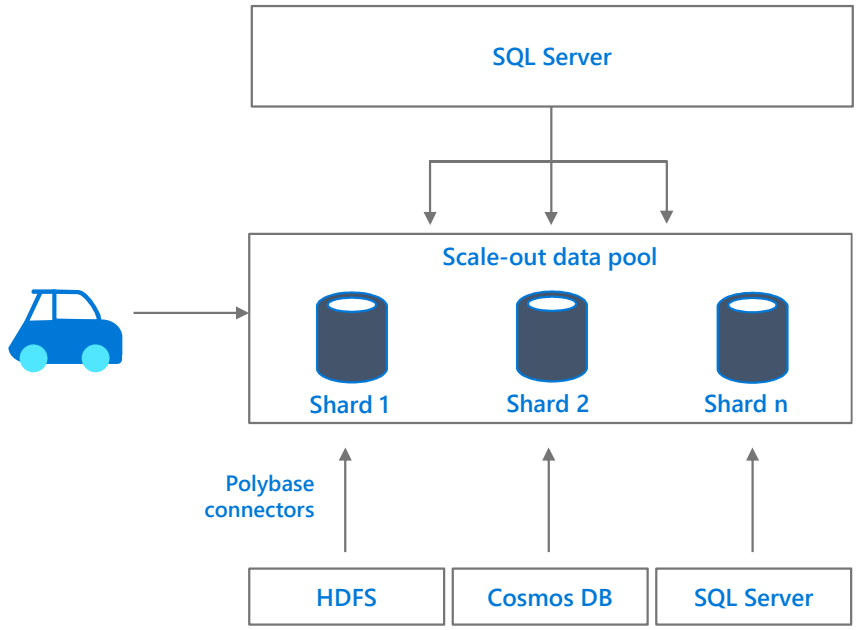
SQL Server 2019 - Data Integration Hub



https://nielsberglund.com

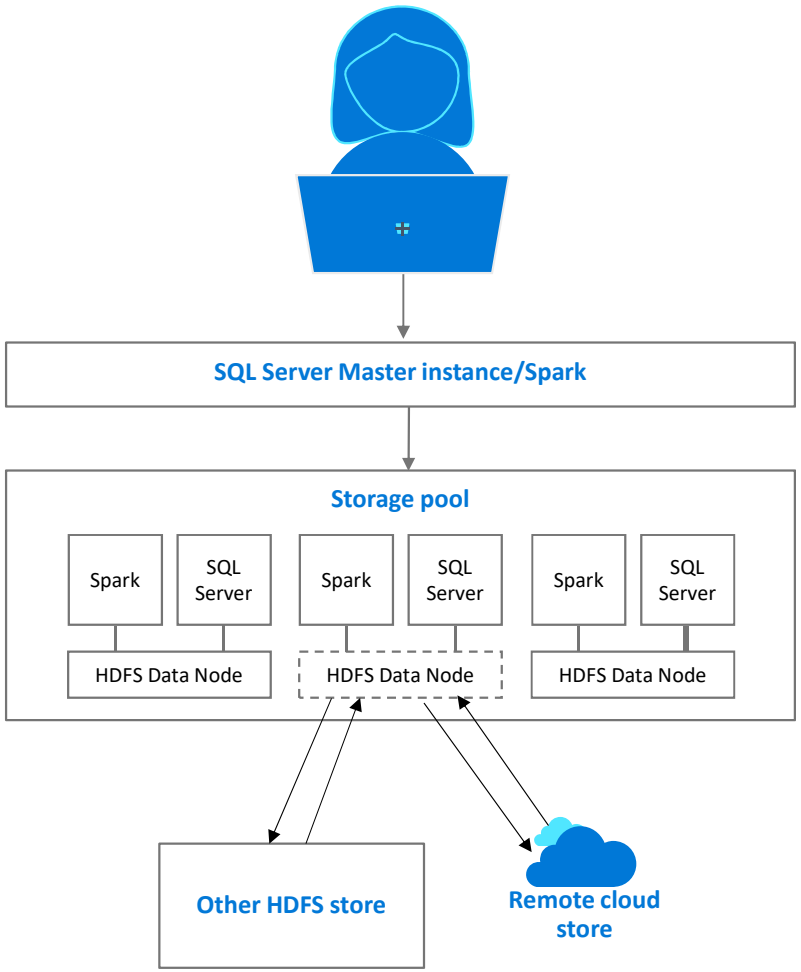
Scale Out - Query Compute

- Query data in relational and non-relational data stores with new PolyBase connectors
- Create a scale-out data pool cache of combined data
- Expose the datasets as a shared data source, **without writing code to move and integrate data**



Scale Out - Storage

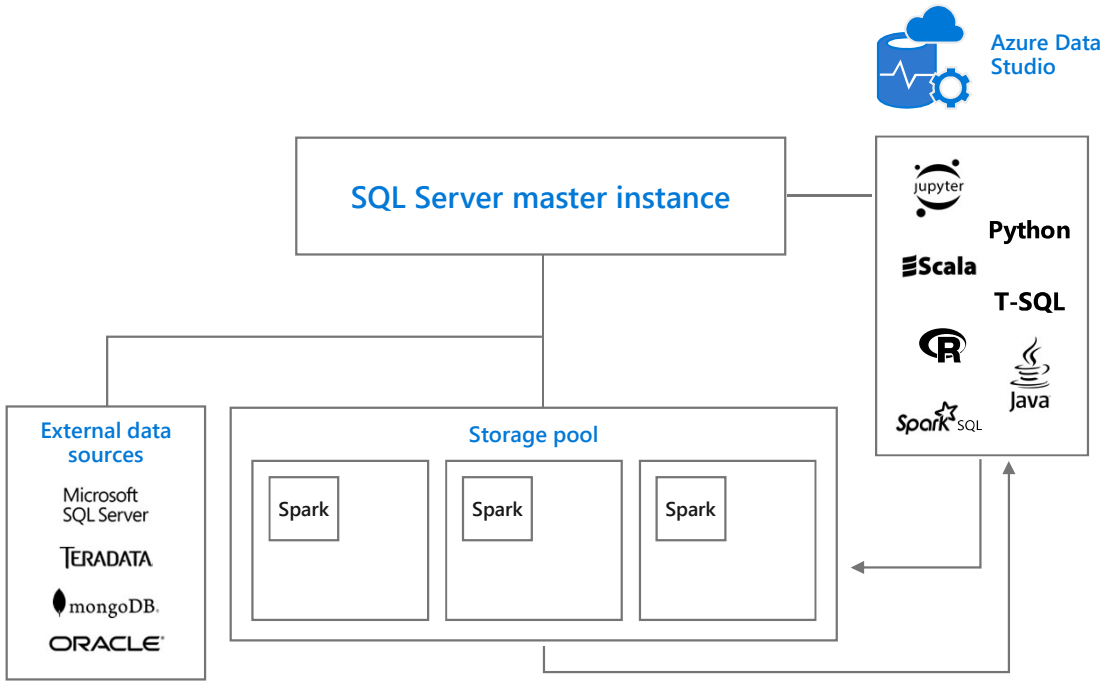
- SQL Server can now read directly from HDFS files.
- Elastically scale compute and storage using HDFS-based storage pools with SQL Server and Spark built in
- Mount and manage remote stores through HDFS
- Mount various on-prem and cloud data stores
- Accelerate computation by caching data locally



https://nielsberglund.com

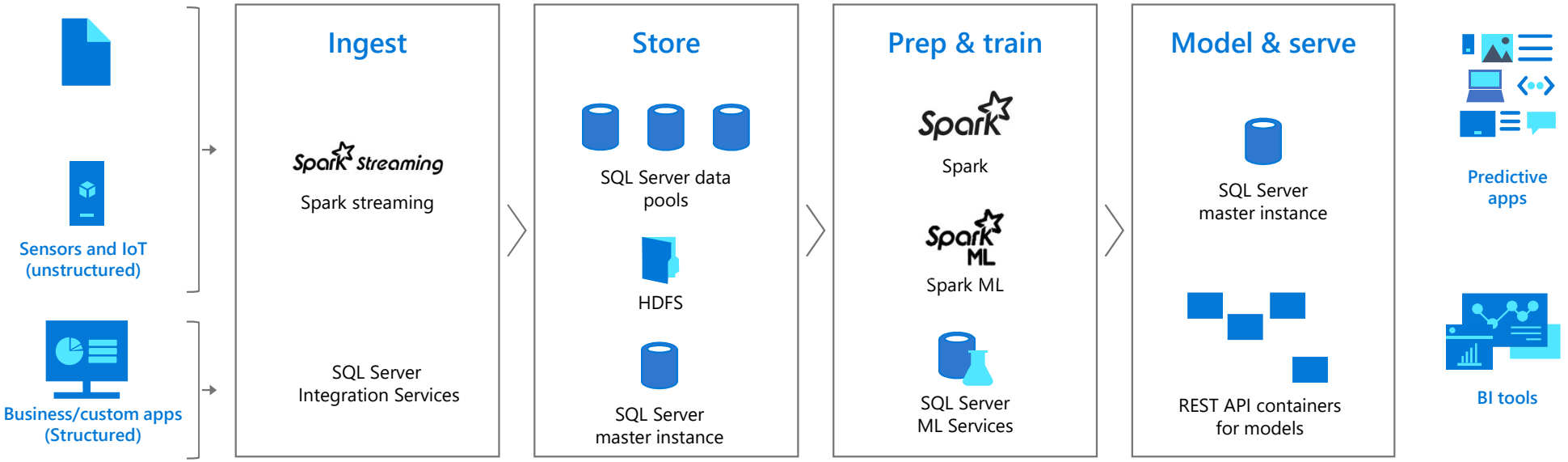
Analyze ALL Data

- Use Azure Data Studio Notebooks to run Spark jobs over structured and unstructured data.
- SPARK SQL can access data in SQL Server.
- Queries can be pushed down to other data sources like Oracle database and Mongo DB.
- Let the Spark job return the data to the notebook.



https://nielsberglund.com

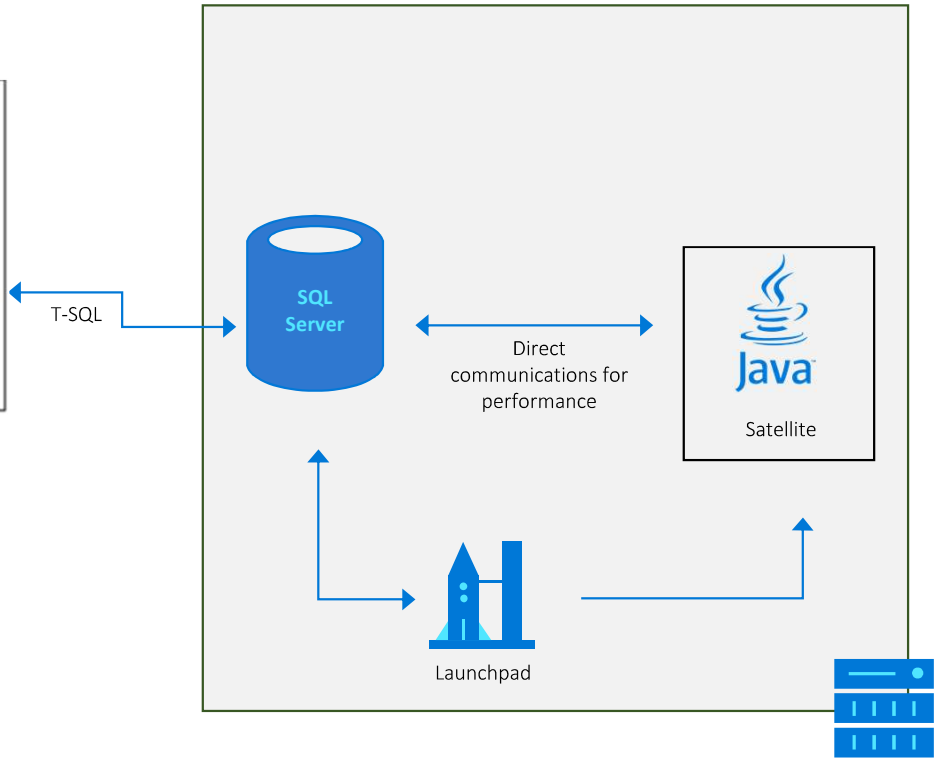
Integrate Structured and Unstructured Data



https://nielsberglund.com

Java Language Extension

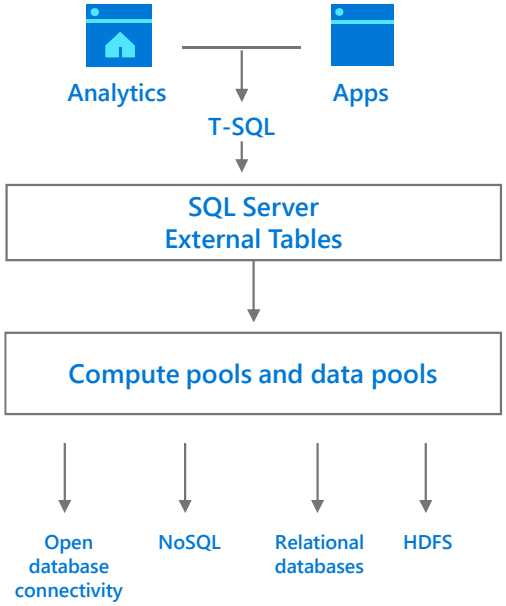
```
DECLARE @myClassPath nvarchar(30)
SET @myClassPath = N'/<my path>/program.jar'
SET @param1 = 3

EXEC sp_execute_external_script
  @language = N'Java'
, @script = N'package.ClassName.MethodName'
, @input_data_1 = N'<Input Query>
, @params = N'@CLASSPATH nvarchar(30), @param1 INT'
, @CLASSPATH = @myClassPath
, @param1 = @param1
with result sets ((outputcol int, outputcol2 int))
```



https://nielsberglund.com

SQL Server Big Data

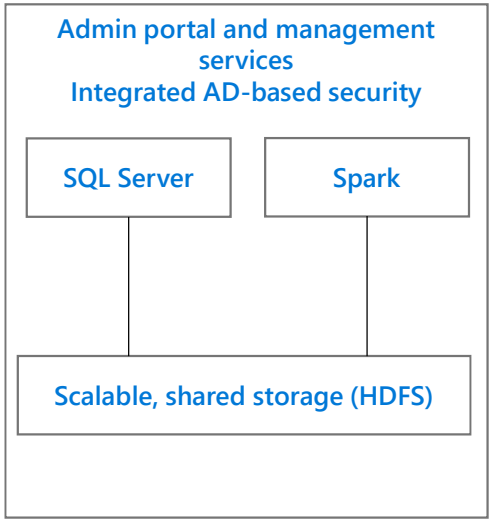
Data virtualization



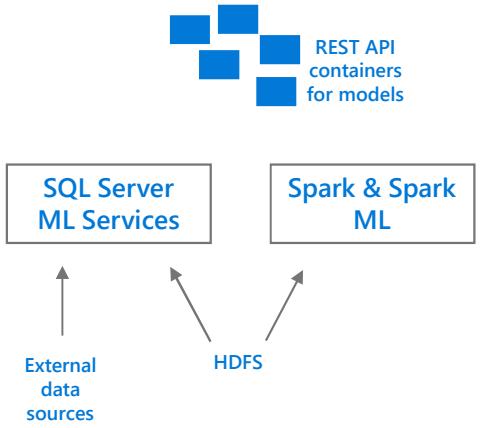
Combine data from many sources without moving or replicating it

Scale out compute and caching to boost performance

Managed SQL Server, Spark and data lake



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

https://nielsberglund.com

Summary

- Data volumes increase by the second.
- The data is of all types and shapes.
- We need a way to easily manage, integrate and handle the data.
- SQL Server 2019 Big Data Cluster runs on Kubernetes.
- Kubernetes:
  - Nodes, Pods, Clusters, Namespace, Volumes.
- SQL Server BDC:
  - Control plane, Master instance, Compute pool, Data pool, Storage pool, App pool.
- Polybase works against more storage types.
- Apache Spark and HDFS part of SQL Server 2019 BDC.

PASS SQLSATURDAY CAPE TOWN | 14 SEP 2019

Thank Sponsors

Thank You!
Questions?

Niels Berglund
niels.it.berglund@gmail.com
https://nielsberglund.com
@nielsberglund