Polybase for SQL Server 2016

Lukasz Grala

Architect Data Platform & BI Solutions | MVP SQL Server

Microsoft

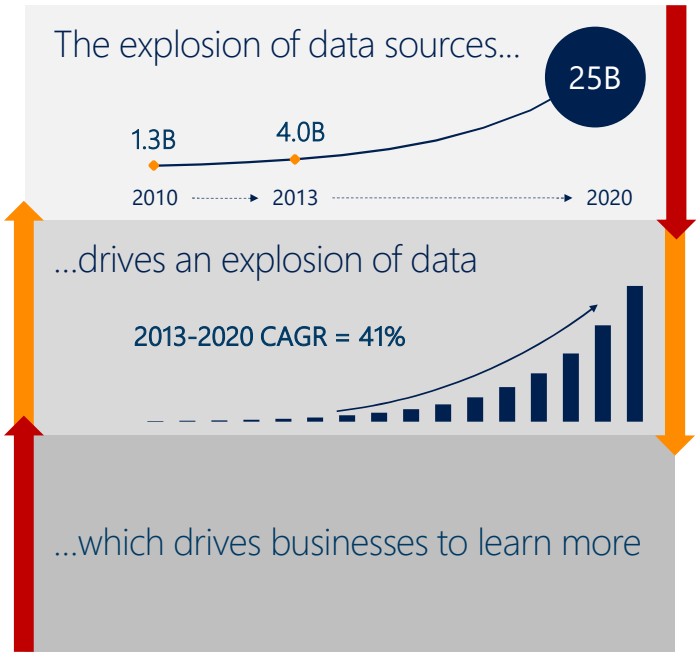
Łukasz Grala

MVP SQL Server | MCT | MCSE

- Architekt i trener - Data Platform & Business Intelligence Solutions
- Prelegent na licznych konferencjach
- Wykładowca i autor publikacji
- Posiada liczne certyfikaty
- Od 2010 roku wyróżniony nagrodą MVP w kategorii SQL Server
- Lider PLSSUG Poznań (lukasz.grala@plssug.org.pl)
- Doktorant na Politechnice Poznańskiej

SQL EXPERT.pl

There's an opportunity to drive smarter decisions with data



Source: Forecast: Internet of Things, Endpoints and Associated Services, Worldwide, 2014. Gartner. Oct 20 2014
Source: IDC "Digital Universe", Dec. 2012

Microsoft platform leads the way on-premises and cloud

Leader in 2014 for Gartner Magic Quadrants

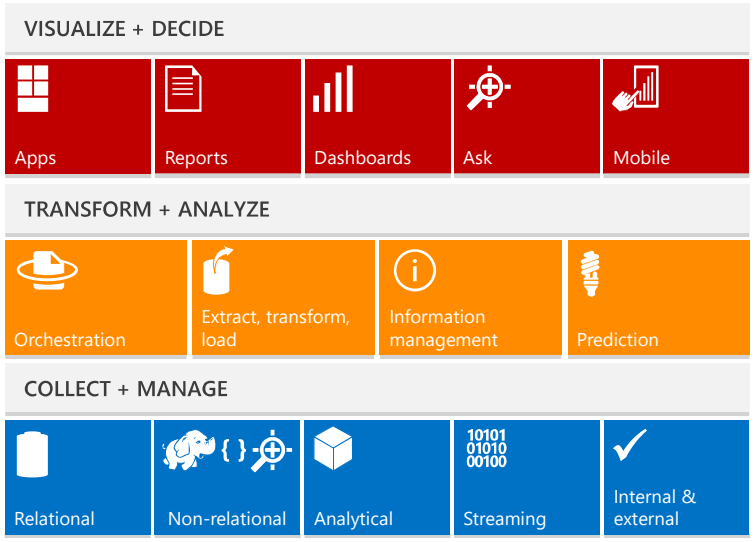
Operational Database Management Systems	Data Warehouse Database Management Systems	Business Intelligence and Analytics Platforms	x86 Server Virtualization	Cloud Infrastructure as a Service	Enterprise Application Platform as a Service	Public Cloud Storage

The Microsoft data platform

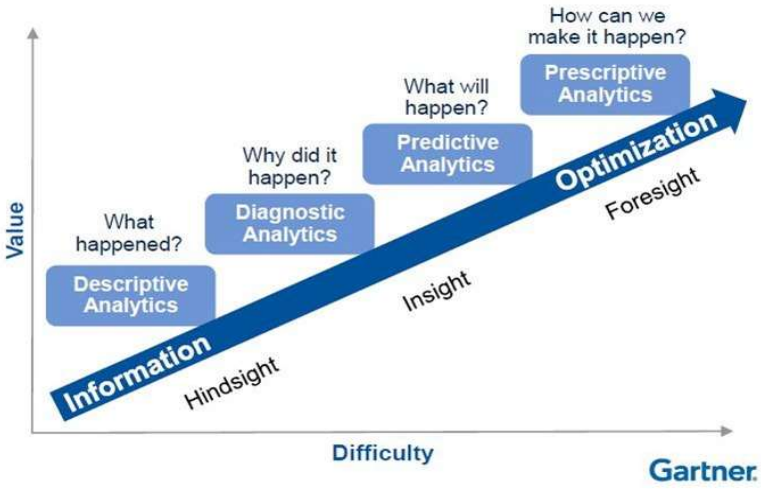
Windows Azure

Office

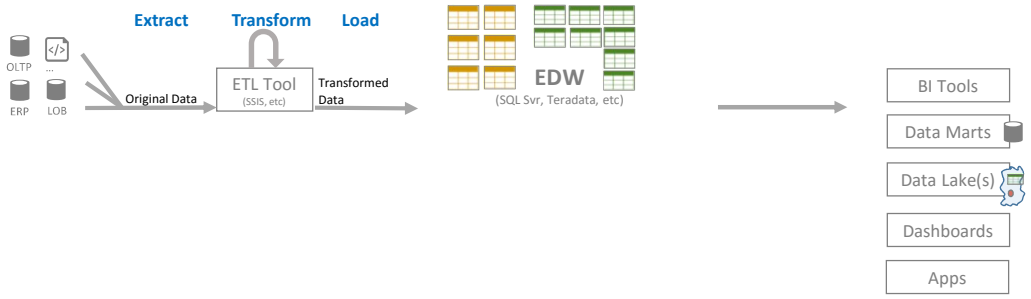
Microsoft SQL Server



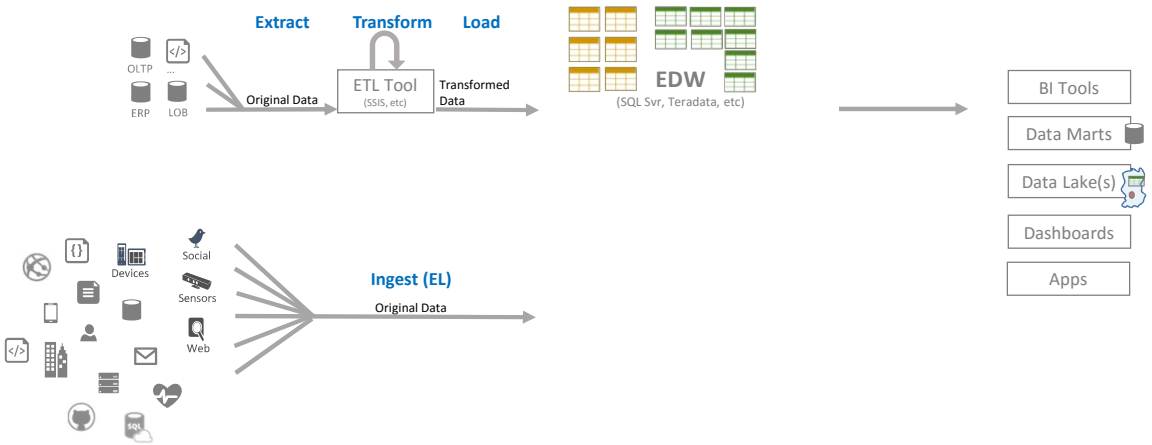
Types of Analytics



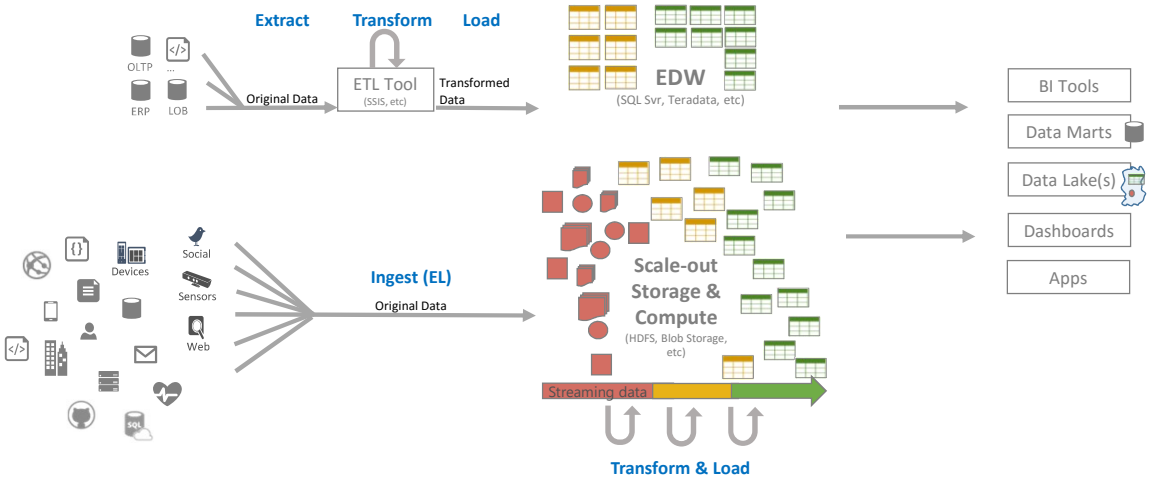
Evolving Approaches to Analytics



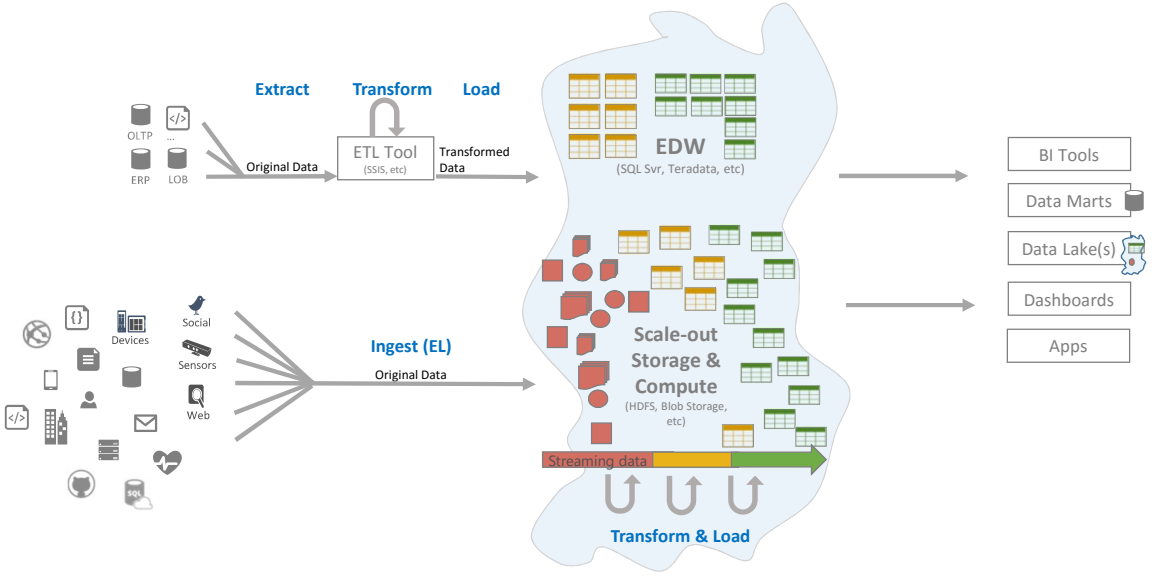
Evolving Approaches to Analytics



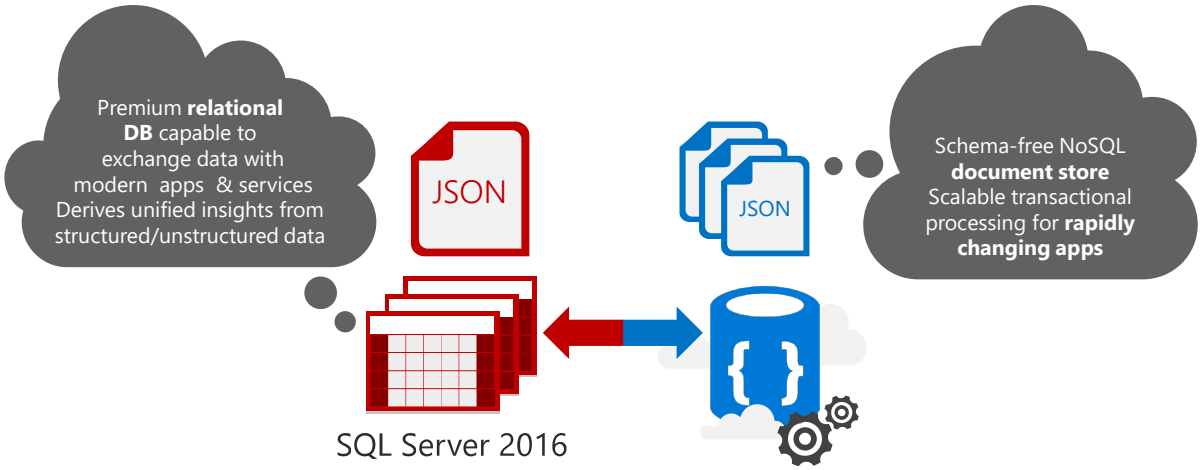
Evolving Approaches to Analytics



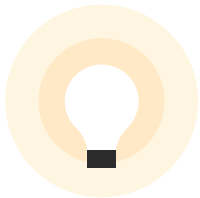
Evolving Approaches to Analytics



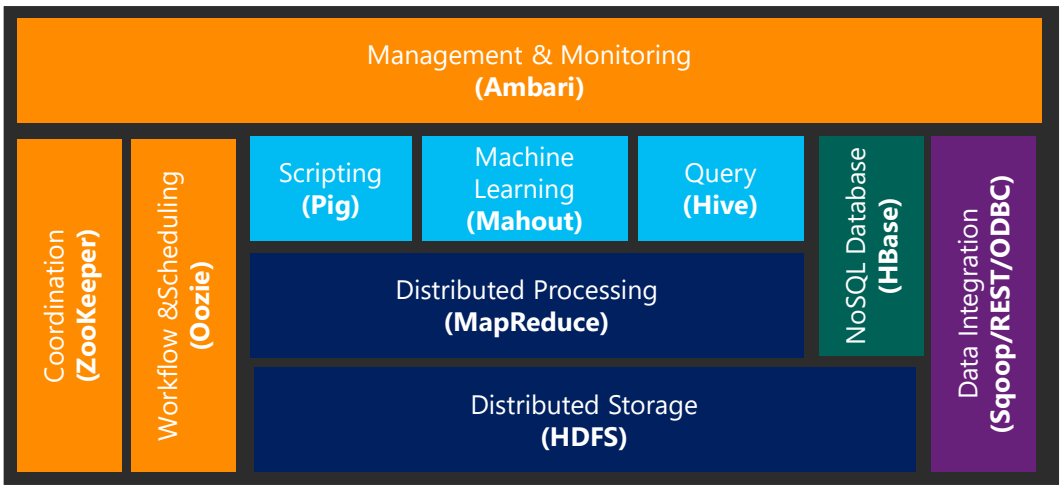
SQL Server and Azure DocumentDB



PolyBase for SQL Server 2016

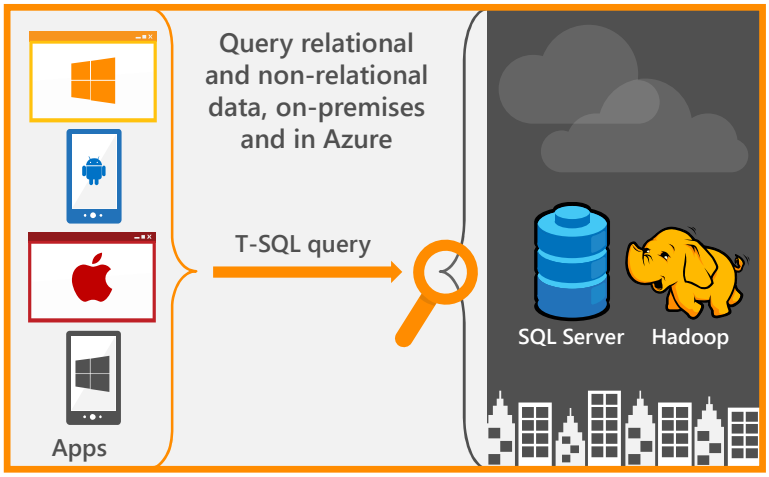


The Hadoop Ecosystem



PolyBase

Query relational and non-relational data with T-SQL



Capability

T-SQL for querying relational and non-relational data across SQL Server and Hadoop

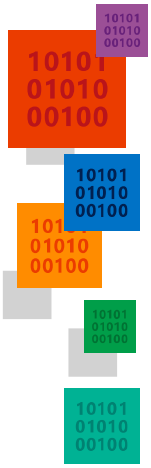
Benefits

New business insights across your data lake

Leverage existing skillsets and BI tools

Faster time to insights and simplified ETL process

Prerequisites for installing PolyBase



64-bit SQL Server Evaluation edition

Microsoft .NET Framework 4.0.

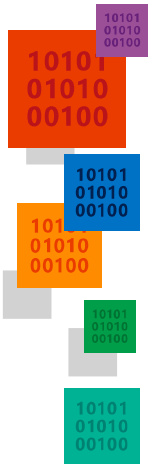
Oracle Java SE RunTime Environment (JRE) version 7.51 or higher

NOTE: Java JRE version 8 does not work.

Minimum memory: 4GB

Minimum hard disk space: 2GB

Using the installation wizard for PolyBase



Run SQL Server Installation Center. (Insert SQL Server installation media and double-click Setup.exe.)

Click Installation, then click New Standalone SQL Server installation or add features

On the feature selection page, select PolyBase Query Service for External Data.

On the Server Configuration Page, configure the PolyBase Engine Service and PolyBase Data Movement Service to run under the same account.

Choose Hadoop data source with sp_configure

```
-- Run sp_configure 'hadoop connectivity'
-- and set an appropriate value
sp_configure
   @configname = 'hadoop connectivity',
   @configvalue = 7;
GO
RECONFIGURE
GO

-- List the configuration settings for
-- one configuration name
sp_configure @configname='hadoop connectivity';
GO
```

Option values

0: Disable Hadoop connectivity

1: Hortonworks HDP 1.3 on Windows Server Azure blob storage (WASB[S])

2: Hortonworks HDP 1.3 on Linux

3: Cloudera CDH 4.3 on Linux

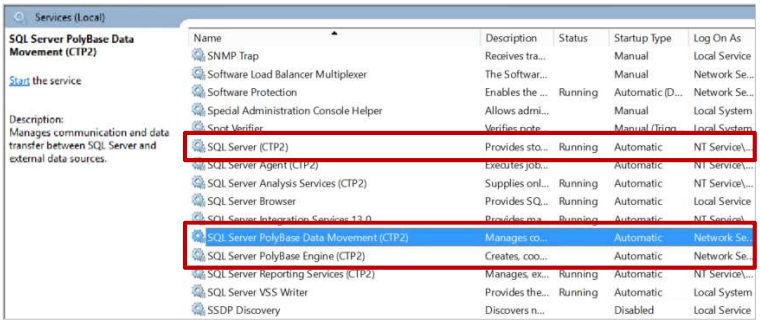
4: Hortonworks HDP 2.0 on Windows Server Azure blob storage (WASB[S])

5: Hortonworks HDP 2.0 on Linux

6: Cloudera 5.1 on Linux

7: Hortonworks 2.1 and 2.2 on Linux
Hortonworks 2.2 on Windows Server Azure blob storage (WASB[S])

Start the PolyBase services



After running for sp_configure, you must stop and restart the SQL Server engine service

Run services.msc

Find the services shown below and stop each one

Restart the services

Configure PolyBase for Azure blob storage

```
-- Using credentials on database requires enabling
-- traceflag
DBCC TRACEON(4631,-1)

-- Create a master key
CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'S0me!nfo';

CREATE CREDENTIAL WASBSecret ON DATABASE WITH
   IDENTITY = 'pdw_user', Secret = 'mykey==';

-- Create an external data source (Azure Blob Storage)
-- with the credential
CREATE EXTERNAL DATA SOURCE Azure_Storage WITH
(  TYPE = HADOOP,
   LOCATION
='wasb[s]://mycontainer@test.blob.core.windows.net/pat
h',
   CREDENTIAL = WASBSecret
)
```

Type methods for providing credentials

Core-site.xml in installation path of SQL Server - <SqlBinRoot>\Polybase\Hadoop\Conf

Credential object in SQL Server for higher security

NOTE: The syntax for a database-scoped credential (CREATE CREDENTIAL … ON DATABASE) is temporary and will change in the next release. This new feature is documented only in the examples in the CTP2 content, and will be fully documented in the next release.

Create a reference to a Hadoop cluster

```
-- Create an external data source (Hadoop)
CREATE EXTERNAL DATA SOURCE hdp2 with (
TYPE = HADOOP,
LOCATION ='hdfs://10.xxx.xx.xxx:xxxx',
RESOURCE_MANAGER_LOCATION='10.xxx.xx.xxx:xxxx')
```

CTP2 supports the following Hadoop distributions

Hortonworks HDP 1.3, 2.0, 2.1, 2.2 for both Windows and Linux

Cloudera CDH 4.3, 5.1 on Linux

Define the external file format

```
-- Create an external file format
-- (delimited text file)
CREATE EXTERNAL FILE FORMAT ff2 WITH (
FORMAT_TYPE = DELIMITEDTEXT,
FORMAT_OPTIONS (FIELD_TERMINATOR ='|',
USE_TYPE_DEFAULT = TRUE))
```

CTP2 supports the following file formats

Delimited text

Hive RCFile

Hive ORC

Create an external table to the data source

```
-- Create an external table pointing to file
stored in Hadoop
CREATE EXTERNAL TABLE [dbo].[CarSensor_Data] (
    [SensorKey] int NOT NULL,
    [CustomerKey] int NOT NULL,
    [GeographyKey] int NULL,
    [Speed] float NOT NULL,
    [YearMeasured] int NOT NULL
)
WITH (LOCATION='/Demo/car_sensordata.tbl',
      DATA_SOURCE = hdp2,
      FILE_FORMAT = ff2,
      REJECT_TYPE = VALUE,
      REJECT_VALUE = 0
```

The external table provides a T-SQL reference to the data source used to:

Query Hadoop or Azure blob storage data with Transact-SQL statements

Import and store data from Hadoop or Azure blob storage into your SQL Server database

Optimize queries by adding statistics

```
-- Create statistics on an external table.
CREATE STATISTICS StatsForSensors ON
CarSensor_Data(CustomerKey, Speed)
```

In CTP2, you can optimize query execution against the external tables using statistics

Query Capabilities (1)

Joining relational and external data

```
SELECT DISTINCT C.FirstName, C.LastName,
C.MaritalStatus
FROM Insurance_Customer_SQL
INNER JOIN (
```

SQL Server table

```
SELECT * FROM SensorData_ExternalHDP WHERE
Speed > 35
UNION ALL
SELECT * FROM SensorData_ExternalHDP2 WHERE
Speed > 35
) AS SensorD
ON C.CustomerKey = SensorD.CustomerKey
```

External tables referring to data in 2 HDP Hadoop clusters

Query Capabilities (2)

Push-Down Computation

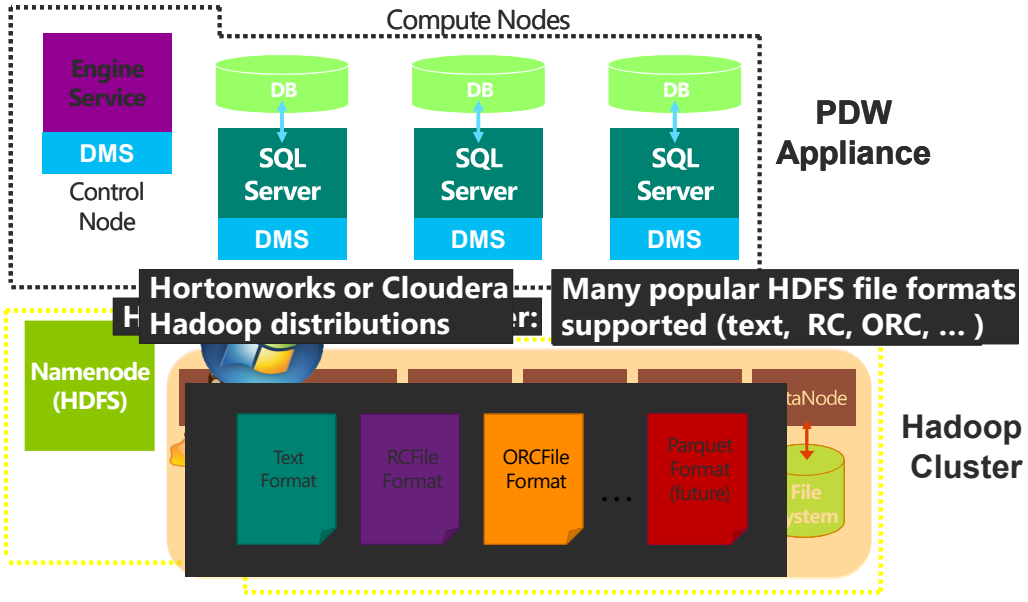
```
SELECT DISTINCT C.FirstName, C.LastName, C.MaritalStatus
FROM Insurance_Customer_SQL -- table in SQL Server
…
OPTION (FORCE EXTERNALPUSHDOWN) - push-down computation
```

```
CREATE EXTERNAL DATA SOURCES ds-hdp WITH .( TYPE = Hadoop,
LOCATION = “hdfs://10.193.27.52:8020”,
Resources_Manager_Location = ‘10.193.27.52:8032’);
```

Pushing Compute

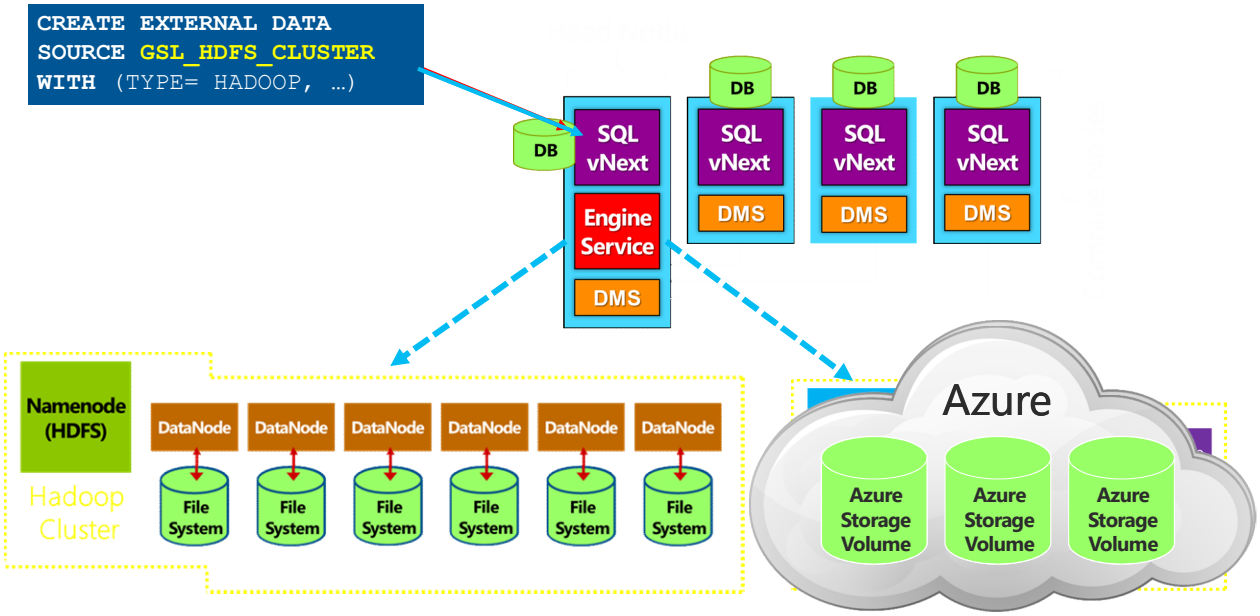
Either on data source level or Per-query basis using new query hints

Typical PolyBase Setup



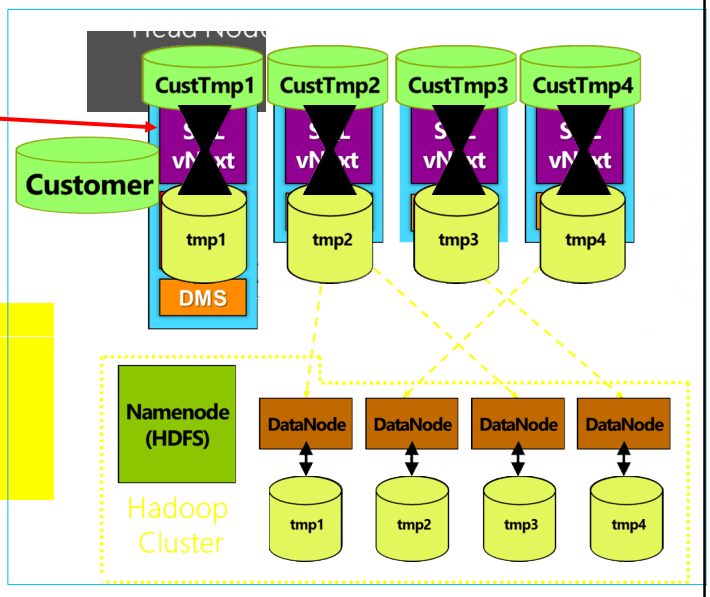
Attach a Hadoop Cluster

```
CREATE EXTERNAL DATA
SOURCE GSL_HDFS_CLUSTER
WITH (TYPE= HADOOP, …)
```

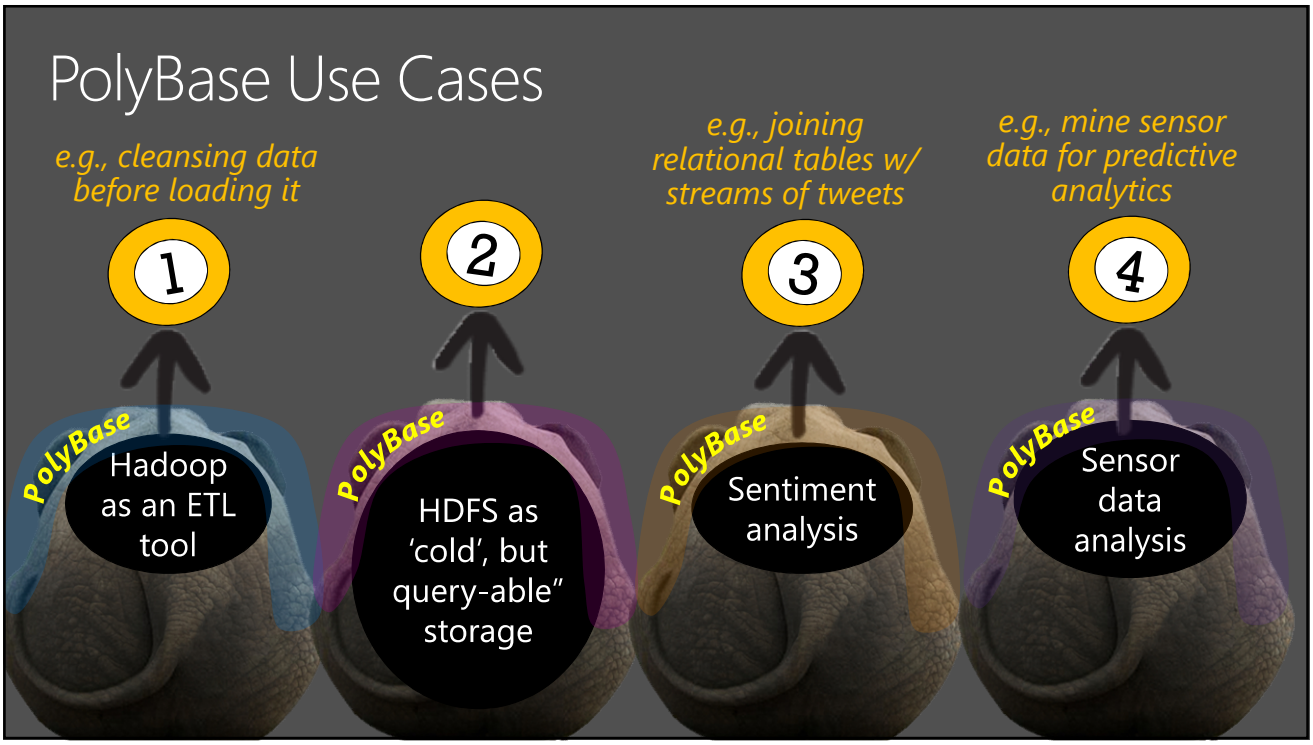


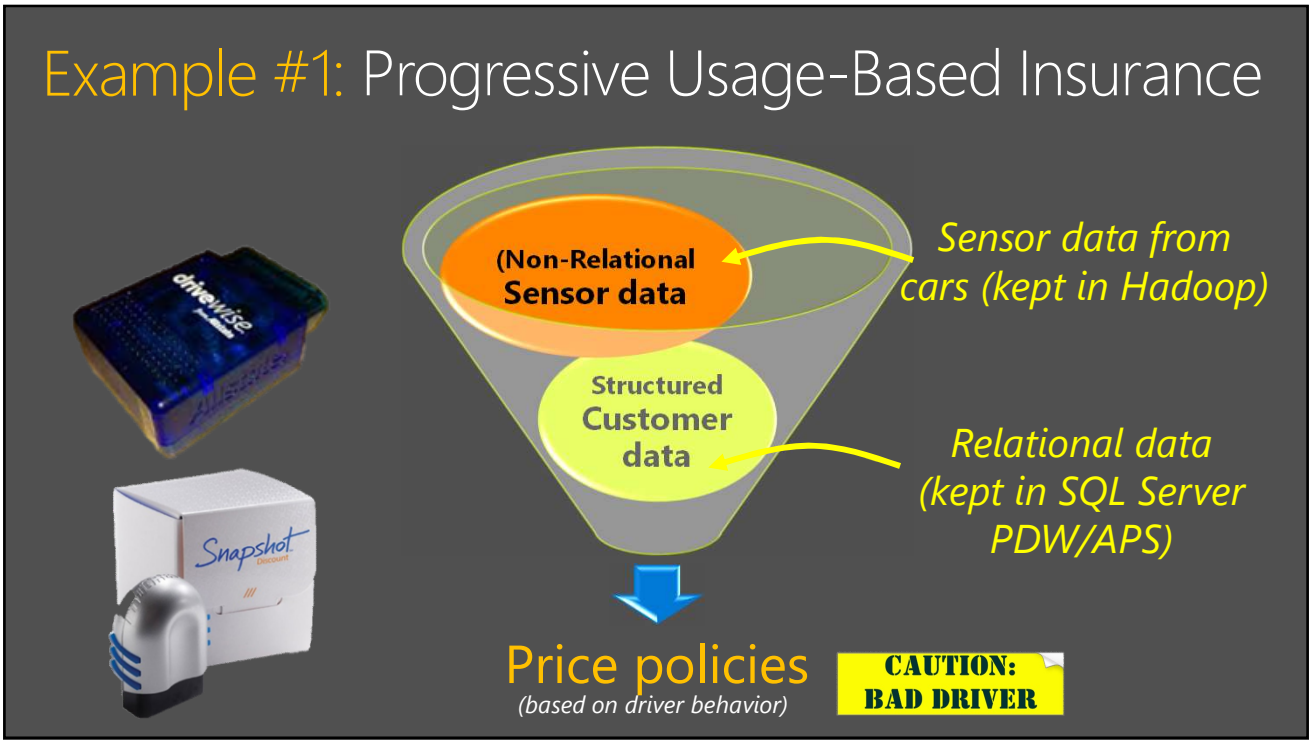
A Better Query Plan

Select C.Name from Customers C, Orders O where C.id = O.CustId and P.price > $1000

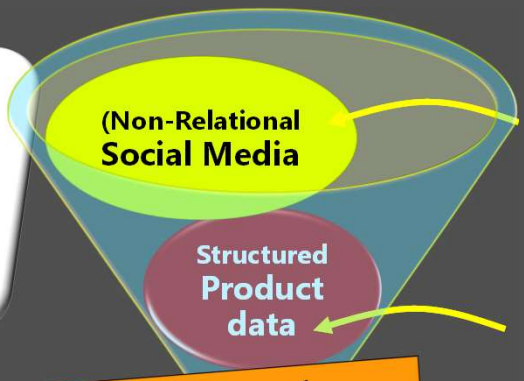


1) Use MR job to find Orders > $1000
2) Import result into SQL vNext instances on Compute nodes
3) **Redistribute Customers table from Head Node to Compute Nodes**
4) Perform join in parallel on compute nodes


PolyBase Use Cases
e.g., cleansing data before loading it
1
PolyBase
Hadoop as an ETL tool
2
PolyBase
HDFS as 'cold', but query-able" storage
e.g., joining relational tables w/ streams of tweets
3
PolyBase
Sentiment analysis
e.g., mine sensor data for predictive analytics
4
PolyBase
Sensor data analysis


Example #1: Progressive Usage-Based Insurance
(Non-Relational Sensor data
Structured Customer data
Sensor data from cars (kept in Hadoop)
Relational data (kept in SQL Server PDW/APS)
Price policies
(based on driver behavior)
CAUTION: BAD DRIVER

Example #2: ShinSeGae (Korea's Amazon)



(kept in Hadoop)

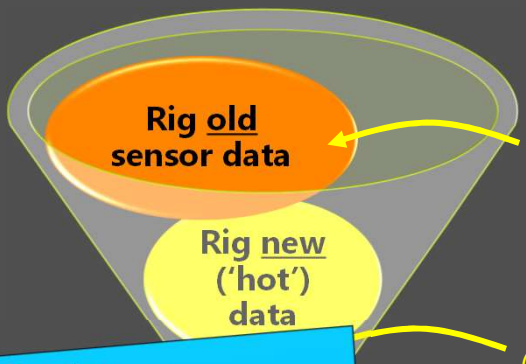
Product data (kept in SQL Server PDW/APS)

All standard BI tools (e.g. Excel, Tableau, Powerview) just work!

Basket Analysis of online shoppers

(based on social media behavior)

Example #3: A Gas & Oil Company



Monitoring & functioning of rig (kept in Hadoop)

More recent data (kept in SQL Server PDW/APS)

Predictive Analysis:
e.g., which rigs are running poorly, which rigs show similar symptoms

alysis

Mobile BI apps for SQL Server – formally Datazen

For on-premises implementations - optimized for SQL Server.

Rich, interactive data visualization on all major mobile platforms

No additional cost for SQL Server Enterprise Edition customers 2008 or later and Software Assurance

Data visualization and publishing

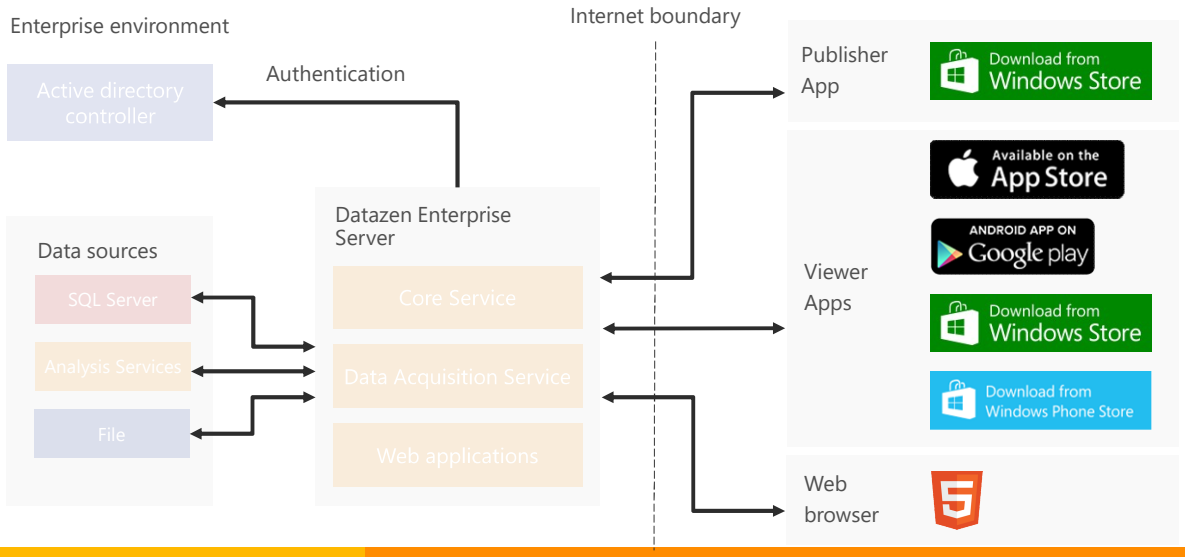
Create beautiful visualizations and KPIs with a touch-based designer

Connect to the Mobile BI for SQL Server server to access SQL Server data

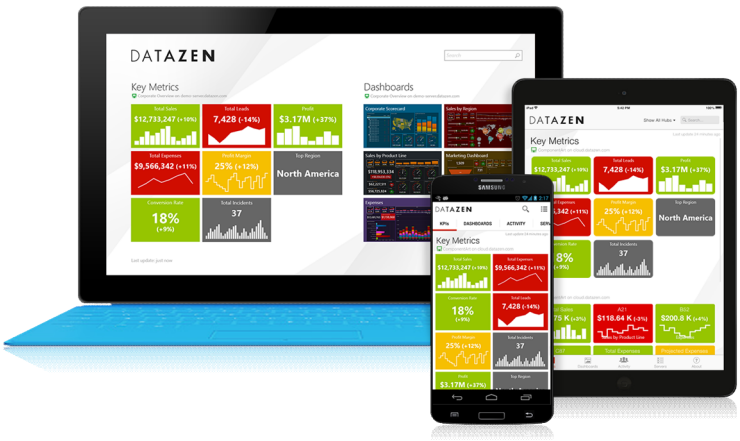
Publish for access by others

Datazen architecture overview

Server, Publisher and Mobile apps



Mobile BI for SQL Server summary



- Beautiful visualizations
- KPI repository
- Create once - publish to any device
- Native apps for all platforms
- Perfect scaling to any form factor
- Custom branding
- Collaborate on the go