Microsoft

Azure Databricks — больше, чем просто Apache Spark на стероидах

Vadim Chelyshkov
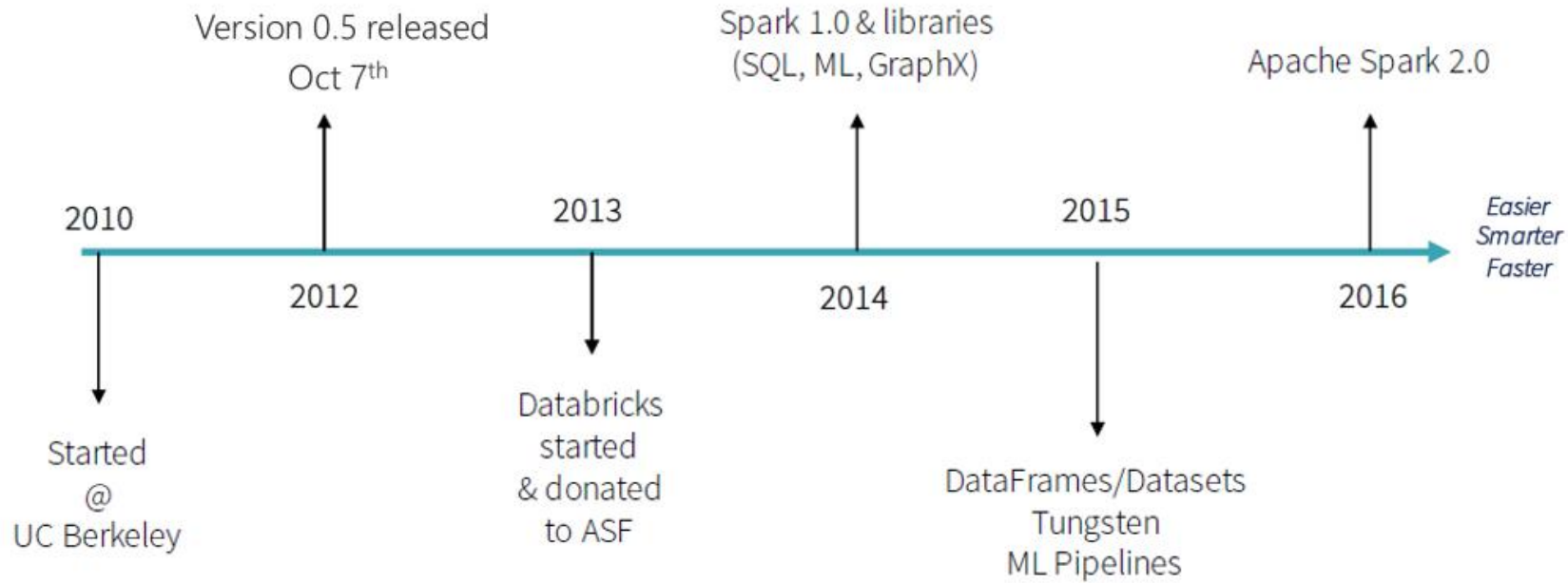Microsoft

Agenda

Apache Spark framework basics

Databricks as a service

Azure DB advantages

Databricks Delta

Apache Spark's brief history

Version 0.5 released
Oct 7th

Spark 1.0 & libraries
(SQL, ML, GraphX)

Apache Spark 2.0

2010

2013

2015

Easier
Smarter
Faster

2012

2014

2016

Started
@
UC Berkeley

Databricks
started
& donated
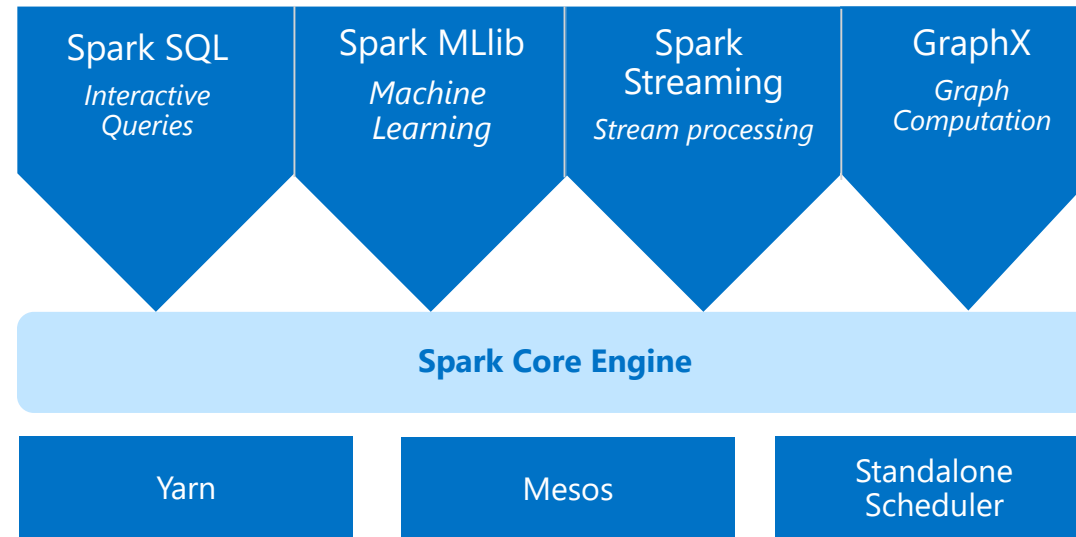to ASF

DataFrames/Datasets
Tungsten
ML Pipelines

Spark framework components

An unified, open source, parallel, data processing framework for Big Data Analytics

Spark Unifies:

- Batch Processing
- Interactive SQL
- Real-time processing
- Machine Learning
- Deep Learning
- Graph Processing

Spark SQL _Interactive Queries_	Spark MLlib _Machine Learning_	Spark Streaming _Stream processing_	GraphX _Graph Computation_

Spark Core Engine

Yarn	Mesos	Standalone Scheduler

Spark benefits

Performance

Using in-memory computing, Spark is considerably faster than Hadoop (100x in some tests).

Can be used for batch and real-time data processing.

Developer Productivity

Easy-to-use APIs for processing large datasets.

Includes 100+ operators for transforming.

Unified Engine

Integrated framework includes higher-level libraries for interactive SQL queries, Stream Analytics, ML and graph processing.

A single application can combine all types of processing

Ecosystem

Spark has built-in support for many data sources, rich ecosystem of ISV applications and a large dev community.

Available on multiple public clouds (AWS, Google and Azure) and multiple on-premises distributors

Cluster architecture

- 'Driver' runs the user's 'main' function and executes the various parallel operations on the worker nodes.

- The results of the operations are collected by the driver

- The worker nodes read and write data from/to Data Sources including HDFS.

- Worker node also cache transformed data in memory as RDDs (Resilient Data Sets).

- Worker nodes and the Driver Node execute as VMs in public clouds (AWS, Google and Azure).
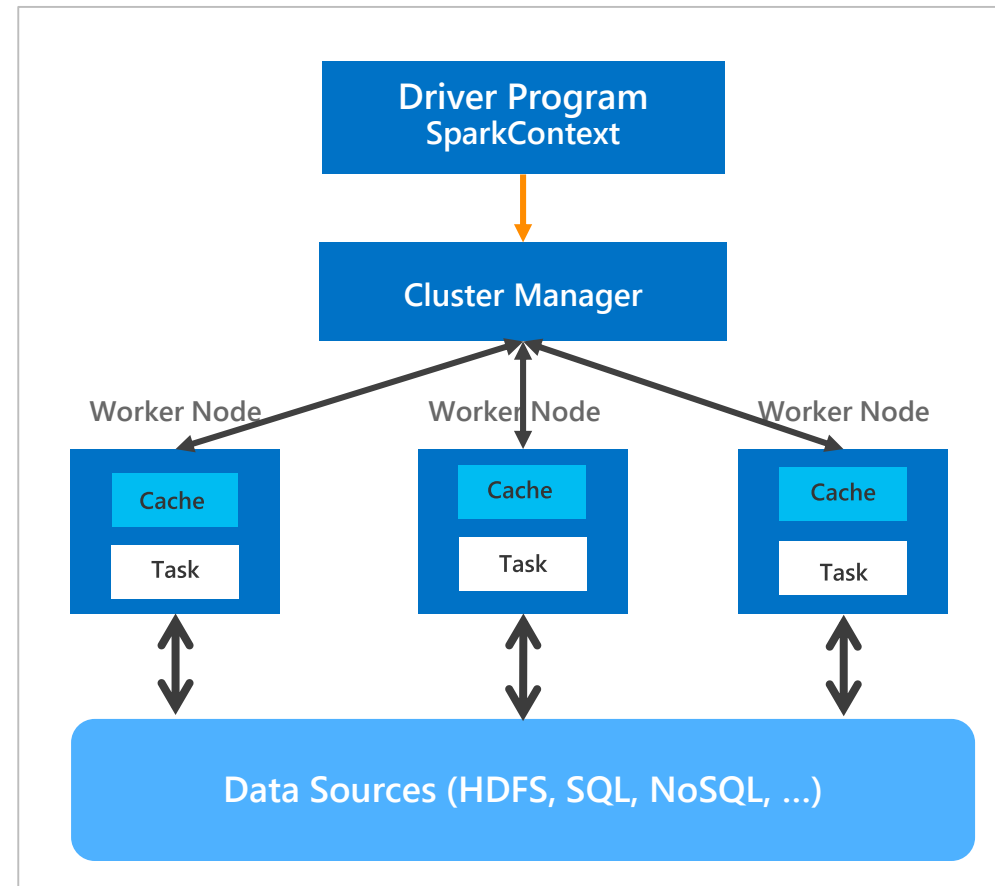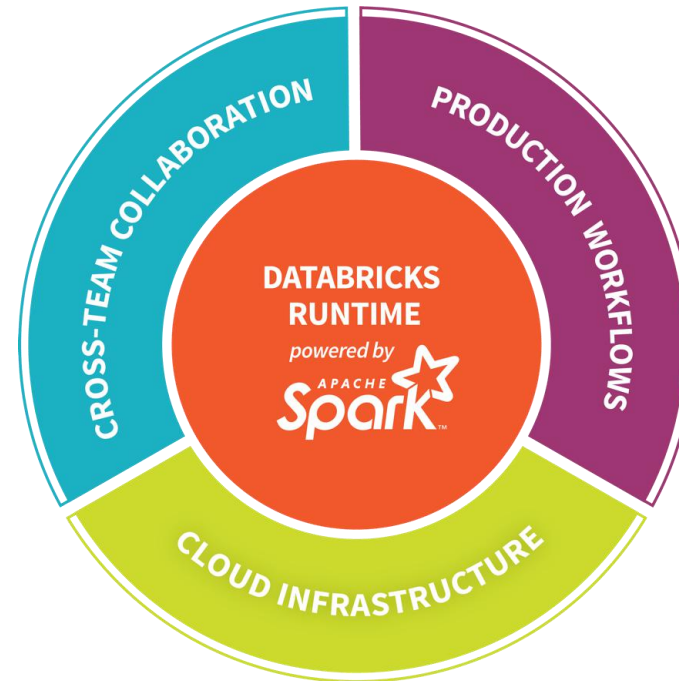
RDD: core, low-level abstraction

- Resilient distributed datasets - fault-tolerant collection of elements stored in memory or on disk that can be *operated on in parallel*

- RDDs support two types of operations: Transformation and Actions

- Transformations create an new dataset from an existing dataset
  - All transformations are lazy: they do not compute their results right away. The transformations are only computed when an action requires a result to be returned to the driver program. Obviously *does not apply to persistent RDDs*.

- Actions return a value to the driver program after running a computation on the dataset.
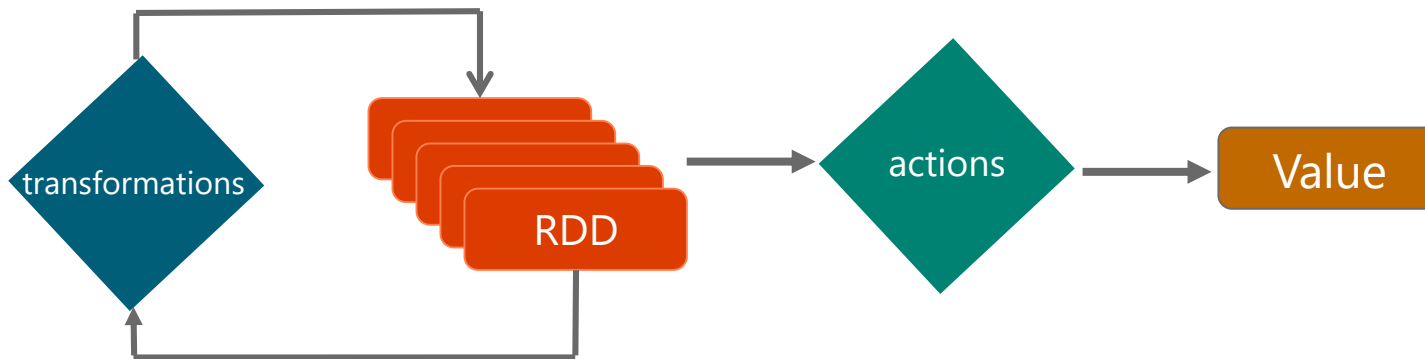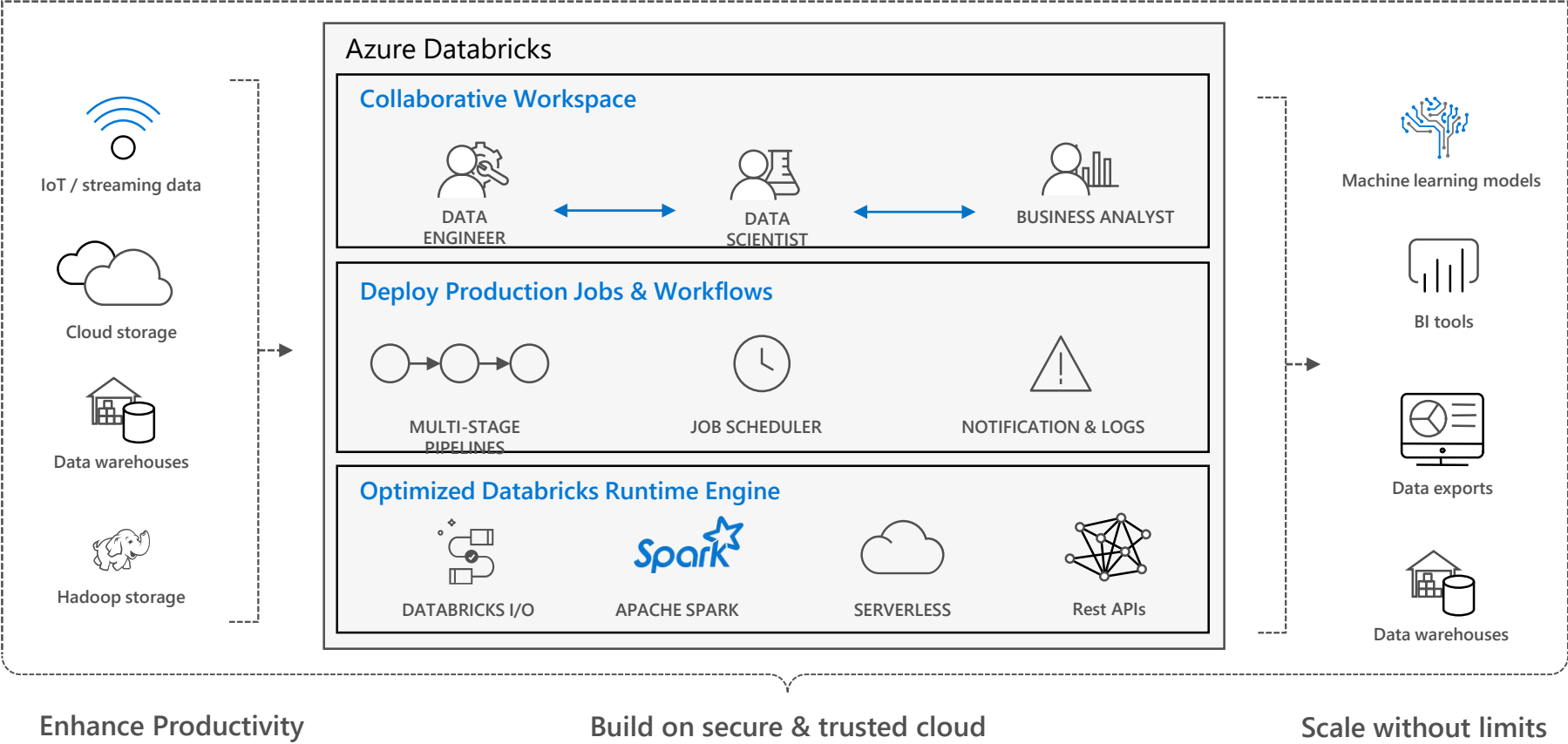
Databricks as a company

- Founded in late 2013

- By the creators of Apache Spark, original team from UC Berkeley AMPLab

- Largest code contributor code to Apache Spark

- Level 2/3 support partnership with
  - Hortonworks
  - MapR
  - DataStax

- Provides certifications such as Databricks Certified Application, Databricks Certified Distribution and Databricks Certified Developer

- Main Product: The Unified Analytics Platform

- In Oct 2017, introduced Databricks Delta (currently in private preview).

Databricks as a platform

Azure Databricks

Collaborative Workspace

DATA ENGINEER ⟷ DATA SCIENTIST ⟷ BUSINESS ANALYST

Deploy Production Jobs & Workflows

MULTI-STAGE PIPELINES   JOB SCHEDULER   NOTIFICATION & LOGS

Optimized Databricks Runtime Engine

DATABRICKS I/O   APACHE SPARK   SERVERLESS   Rest APIs

IoT / streaming data

Cloud storage

Data warehouses

Hadoop storage

Machine learning models

BI tools

Data exports

Data warehouses

Enhance Productivity   **Build on secure & trusted cloud**   **Scale without limits**

Collaborative Workspace

GET STARTED IN SECONDS

Single click to launch your new Spark environment

INTERACTIVE EXPLORATION

Explore data using interactive notebooks with support for multiple programming languages including R, Python, Scala, and SQL

COLLABORATION

Work on the same notebook in real-time while tracking changes with detailed revision history, GitHub, or Bitbucket

VISUALIZATIONS

Visualize insights through a wide assortment of point-and-click visualizations. Or use powerful scriptable options like matplotlib, ggplot, and D3

DASHBOARDS

Rich integration with PowerBI to discover and share your insights in powerful new ways
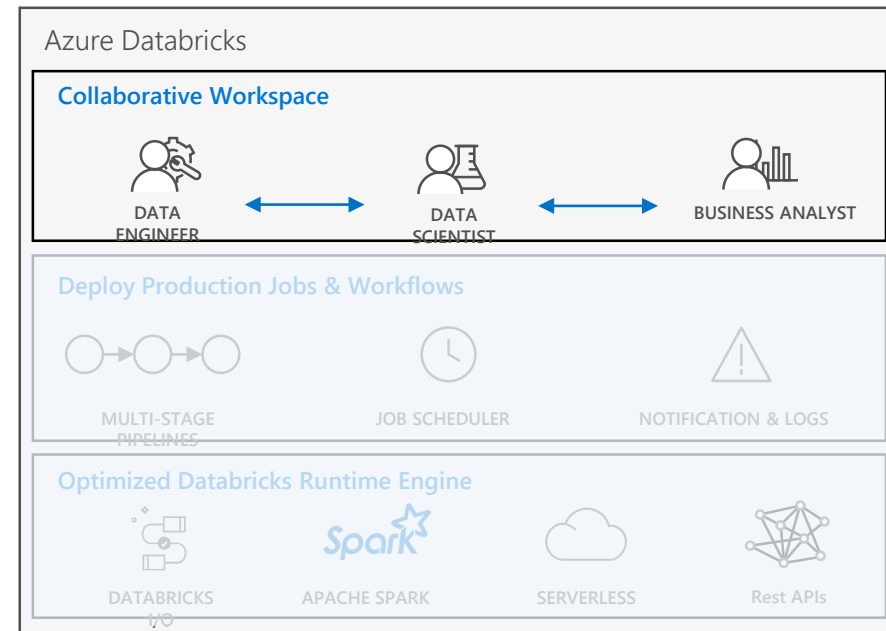
Deploy Production Jobs & Workflows

JOBS SCHEDULER

Execute jobs for production pipelines on a specific schedule

NOTEBOOK WORKFLOWS

Create multi-stage pipelines with the control structures of the source programming language
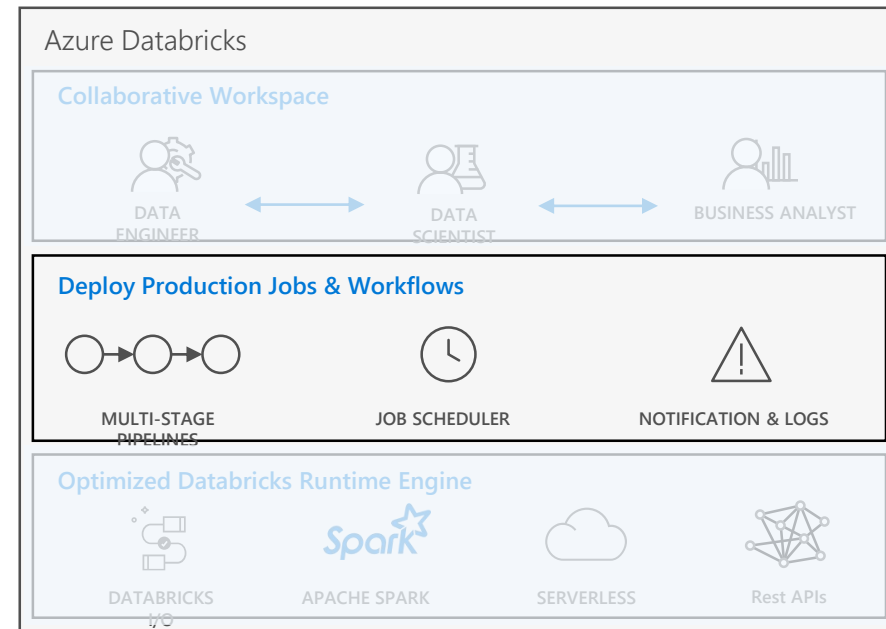
RUN NOTEBOOKS AS JOBS

Turn notebooks or JARs into resilient Spark jobs with a click or an API call

NOTIFICATIONS AND LOGS

Set up alerts and quickly access audit logs for easy monitoring and troubleshooting

INTEGRATE NATIVELY WITH AZURE SERVICES

Deep integration with Azure SQL Data Warehouse, Cosmos DB, Azure Data Lake Store, Azure Blob Storage, and Azure Event Hub

Azure Databricks

Collaborative Workspace

DATA ENGINEER — DATA SCIENTIST — BUSINESS ANALYST

Deploy Production Jobs & Workflows

MULTI-STAGE PIPELINES | JOB SCHEDULER | NOTIFICATION & LOGS

Optimized Databricks Runtime Engine

DATABRICKS I/O | APACHE SPARK | SERVERLESS | Rest APIs

Optimized Databricks Runtime Engine

OPTIMIZED I/O PERFORMANCE

The Databricks I/O module (DBIO) takes processing speeds to the next level — significantly improving the performance of Spark in the cloud
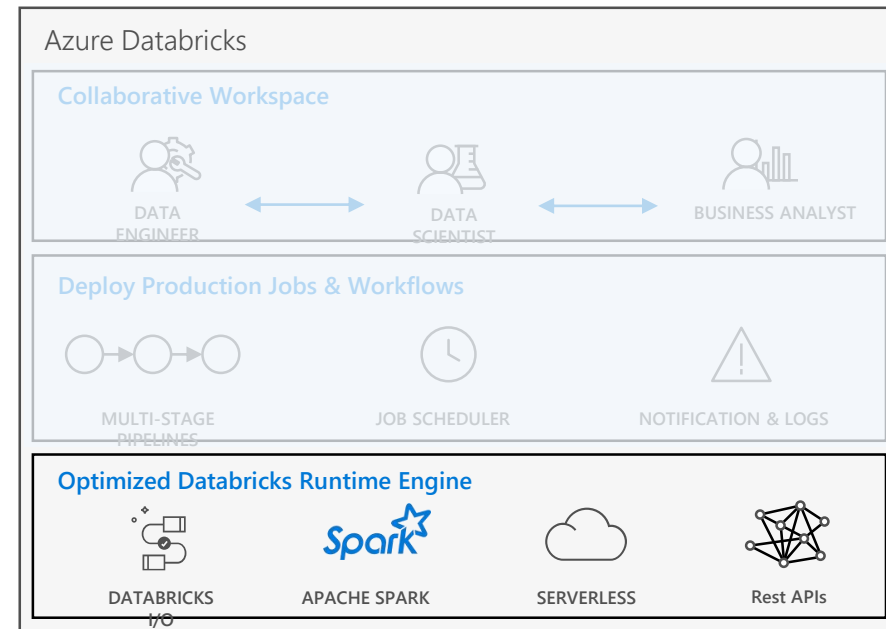
FULLY-MANAGED PLATFORM ON AZURE

Reap the benefits of a fully managed service and remove the complexity of big data and machine learning

SERVERLESS INFRASTRUCTURE

Databricks' serverless and highly elastic cloud service is designed to remove operational complexity while ensuring reliability and cost efficiency at scale
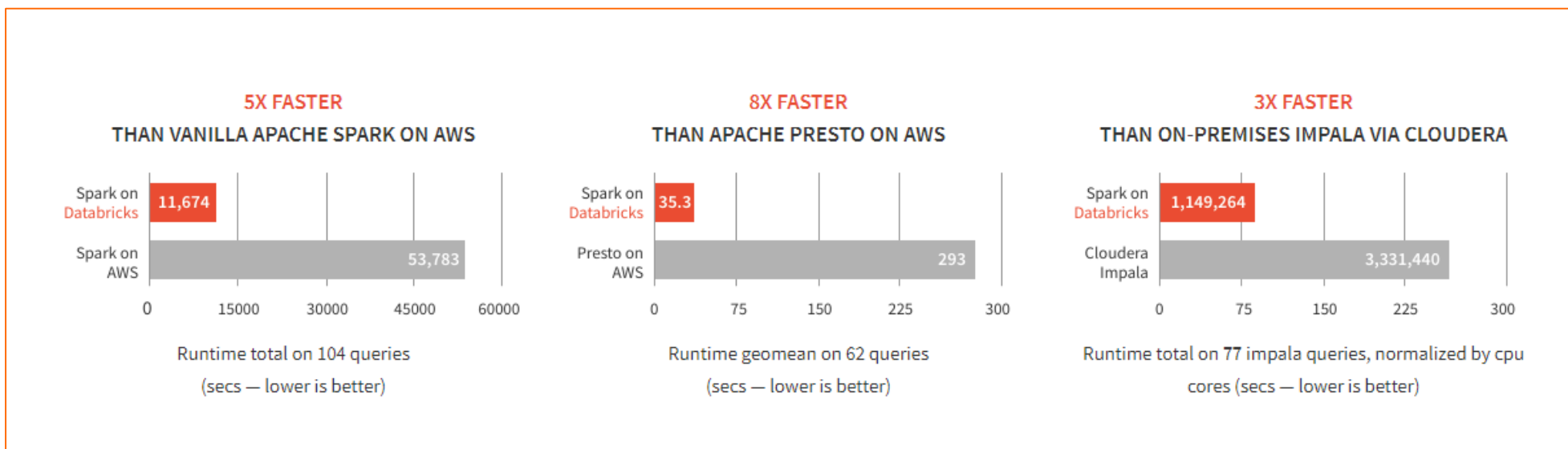
OPERATE AT MASSIVE SCALE

Without limits globally
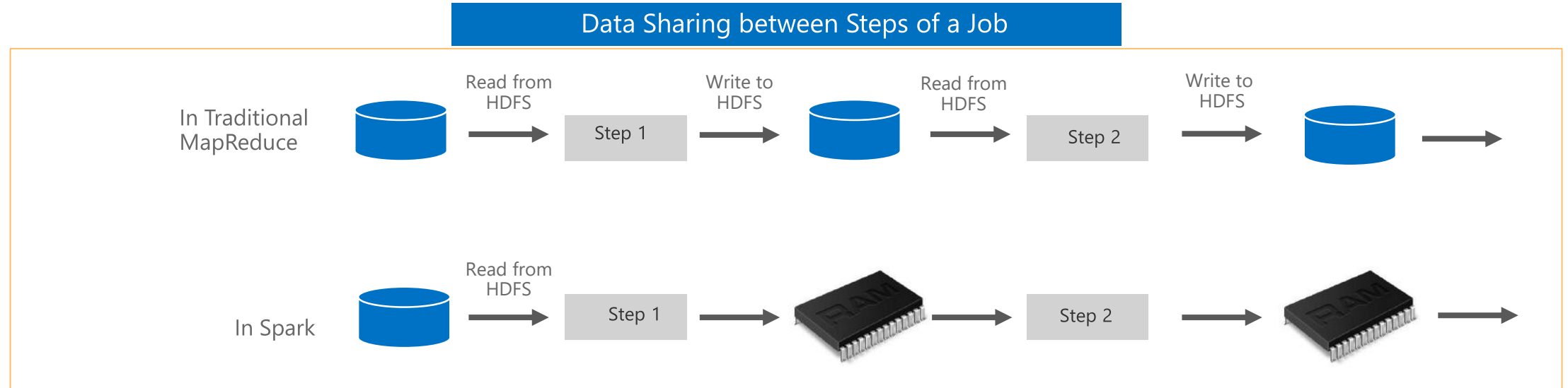

Azure Databricks

Collaborative Workspace

DATA ENGINEER ↔ DATA SCIENTIST ↔ BUSINESS ANALYST

Deploy Production Jobs & Workflows

MULTI-STAGE PIPELINES   JOB SCHEDULER   NOTIFICATION & LOGS

Optimized Databricks Runtime Engine

DATABRICKS I/O   APACHE SPARK   SERVERLESS   Rest APIs

Demo

Databricks workspace

DATABRICKS SPARK IS FAST

Benchmarks have shown Databricks to often have better performance than alternatives

5X FASTER
THAN VANILLA APACHE SPARK ON AWS

Spark on Databricks	11,674
Spark on AWS	53,783

Runtime total on 104 queries
(secs — lower is better)

8X FASTER
THAN APACHE PRESTO ON AWS

Spark on Databricks	35.3
Presto on AWS	293

Runtime geomean on 62 queries
(secs — lower is better)

3X FASTER
THAN ON-PREMISES IMPALA VIA CLOUDERA

Spark on Databricks	1,149,264
Cloudera Impala	3,331,440

Runtime total on 77 impala queries, normalized by cpu
cores (secs — lower is better)

SOURCE: Benchmarking Big Data SQL Platforms in the Cloud

- **In-memory cluster computing:** Spark provides primitives for *in-memory* cluster computing. A Spark job can *load and cache* data into memory and query it repeatedly (iteratively) much quicker than disk-based systems.

- **Scala Integration:** Spark integrates into the Scala programming language, letting you manipulate distributed datasets like local collections. No need to structure everything as map and reduce operations

- **Faster Data-sharing:** Data-sharing between operations is faster as data is in-memory:

  - In (traditional) Hadoop data is shared through HDFS which is expensive. HDFS maintains three replicas.
  - Spark stores data in-memory *without any replication*.

Data Sharing between Steps of a Job

In Traditional MapReduce — Read from HDFS → Step 1 → Write to HDFS → Read from HDFS → Step 2 → Write to HDFS →

In Spark — Read from HDFS → Step 1 → RAM → Step 2 → RAM →

Databricks IO Cache automatically caches 'remote' data on 'local nodes' to accelerate data reads

- A copy of the remote file is created in the node's local storage
  - Local data is stored in a fast intermediate format
  - Currently *Parquet* file format is supported

- Remote data is cached automatically

- Supports *DBFS, HDFS, Azure Blob Storage and Azure Data Lake store*

- DBIO Cache lets you:
  - Enable or disable caching at anytime
  - Cache only a select subset of the data

- DBIO Cache has to be configured during cluster creation. The '*max disk space per node reserved for cached data*' must be specified during cluster creation

You can Monitor the state of the DBIO cache in the Portal

Storage

Parquet IO Cache

Host	Disk Usage	Max Disk Usage Limit	Percent Disk Usage	Metadata Cache Size	Max Metadata Cache Size Limit	Percent Metadata Usage
10.0.185.226	8.3 GB	442.4 GB	1 %	6.8 MB	8.8 GB	0 %
10.0.194.201	8.2 GB	442.4 GB	1 %	6.8 MB	8.8 GB	0 %
10.0.199.229	8.2 GB	442.4 GB	1 %	6.9 MB	8.8 GB	0 %
10.0.215.147	8.1 GB	442.4 GB	1 %	7.0 MB	8.8 GB	0 %
Total	32.8 GB	1769.5 GB	1 %	27.4 MB	35.4 GB	0 %

RDDS AND DBIO CACHE - DIFFERENCES

DBIO cache and RDDs are both caches that can be used together

Capability	Comment
Availability	• RDD is part of Apache Spark • Databricks IO cache is available only to Databricks customers.
Type of data stored	• The RDD cache can be used to store the result of any subquery. • The DBIO cache is designed to speed-up scans by creating local copies of remote data. It can improve the performance of a wide range of queries, but cannot be used to store results of arbitrary subqueries.
Performance	• The data stored in the DBIO cache can be read and operated on faster than the data in the RDD cache. This is because the DBIO cache uses efficient decompression algorithms, and outputs data in the optimal format for further processing using whole-stage code generation.
Automatic vs manual control	• When using the RDD cache it is necessary to manually choose tables or queries to be cached. • When using the DBIO cache the data is added to the cache automatically whenever it has to be fetched from a remote source. This process is fully transparent and does not require any action from the user.
Disk vs memory-based	• Unlike the RDD cache, the DBIO cache is stored entirely on the local disk.

Databricks loves Azure

Integration capabilities

AZURE BLOB STORAGE INTEGRATION

Data can be read from Azure Blob Storage using the Hadoop FileSystem interface. Data can be read from public storage accounts without any additional settings. To read data from a private storage account, you need to set an account key or a Shared Access Signature (SAS) in your notebook

Setting up an account key

```
spark.conf.set ( "fs.azure.account.key.{Your Storage Account Name}.blob.core.windows.net", "{Your Storage Account Access Key}")
```

Setting up a SAS for a given container:

```
spark.conf.set( "fs.azure.sas.{Your Container Name}.{Your Storage Account Name}.blob.core.windows.net", "{Your SAS For The Given Container}")
```

Once an account key or a SAS is setup, you can use standard Spark and Databricks APIs to read from the storage account:

```
val df = spark.read.parquet("wasbs://{Your Container Name}@m{Your Storage Account  name}.blob.core.windows.net/{Your Directory Name}")
dbutils.fs.ls("wasbs://{Your ntainer Name}@{Your Storage Account Name}.blob.core.windows.net/{Your Directory Name}")
```

AZURE DATA LAKE INTEGRATION

To read from your Data Lake Store account, you can configure Spark to use service credentials with the following snippet in your notebook

```
spark.conf.set("dfs.adls.oauth2.access.token.provider.type", "ClientCredential")

spark.conf.set("dfs.adls.oauth2.client.id", "{YOUR SERVICE CLIENT ID}")

spark.conf.set("dfs.adls.oauth2.credential", "{YOUR SERVICE CREDENTIALS}")

spark.conf.set("dfs.adls.oauth2.refresh.url", "https://login.windows.net/{YOUR DIRECTORY ID}/oauth2/token")
```

After providing credentials, you can read from Data Lake Store using standard APIs:

```
val df = spark.read.parquet("adl://{YOUR DATA LAKE STORE ACCOUNT NAME}.azuredatalakestore.net/{YOUR DIRECTORY NAME}")

dbutils.fs.list("adl://{YOUR DATA LAKE STORE ACCOUNT NAME}.azuredatalakestore.net/{YOUR DIRECTORY NAME}")
```

COSMOS DB INTEGRATION

The Spark connector enables real-time analytics over globally distributed data in Azure Cosmos DB

Azure Cosmos DB is Microsoft's globally distributed, multi-model database service for mission-critical applications

- With Spark connector for Azure Cosmos DB, Apache Spark can now interact with all Azure Cosmos DB data models: *Documents, Tables, and Graphs*.
  - efficiently exploits the native Azure Cosmos DB managed indexes and enables updateable columns when performing analytics.
  - utilizes push-down predicate filtering against fast-changing globally-distributed data
- Some use-cases for Azure Cosmos DB + Spark include:
  - Streaming Extract, Transformation, and Loading of data (ETL)
  - Data enrichment
  - Trigger event detection
  - Complex session analysis and personalization
  - Visual data exploration and interactive analysis
  - Notebook experience for data exploration, information sharing, and collaboration

The connector uses the Azure DocumentDB Java SDK and moves data directly between Spark worker nodes and Cosmos DB data nodes
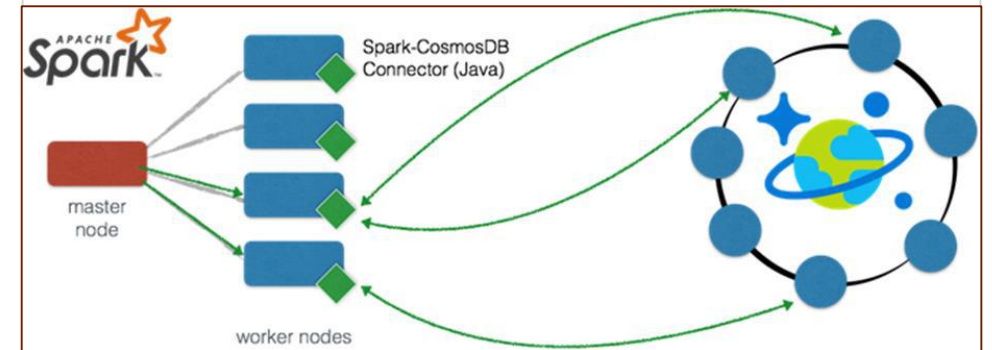
AZURE SQL DW INTEGRATION

Integration enables structured data from SQL DW to be included in Spark Analytics

Azure SQL Data Warehouse is a SQL-based fully managed, petabyte-scale cloud solution for data warehousing

- You can bring in data from Azure SQL DW to perform advanced analytics that require both structured and unstructured data.

- Currently you can access data in Azure SQL DW via the JDBC driver. From within your spark code you can access just like any other JDBC data source.

- If Azure SQL DW is authenticated via AAD then Azure Databricks user can seamlessly access Azure SQL DW.

Spark SQL   Java JDBC

Azure Databricks

Azure

Azure SQL DW
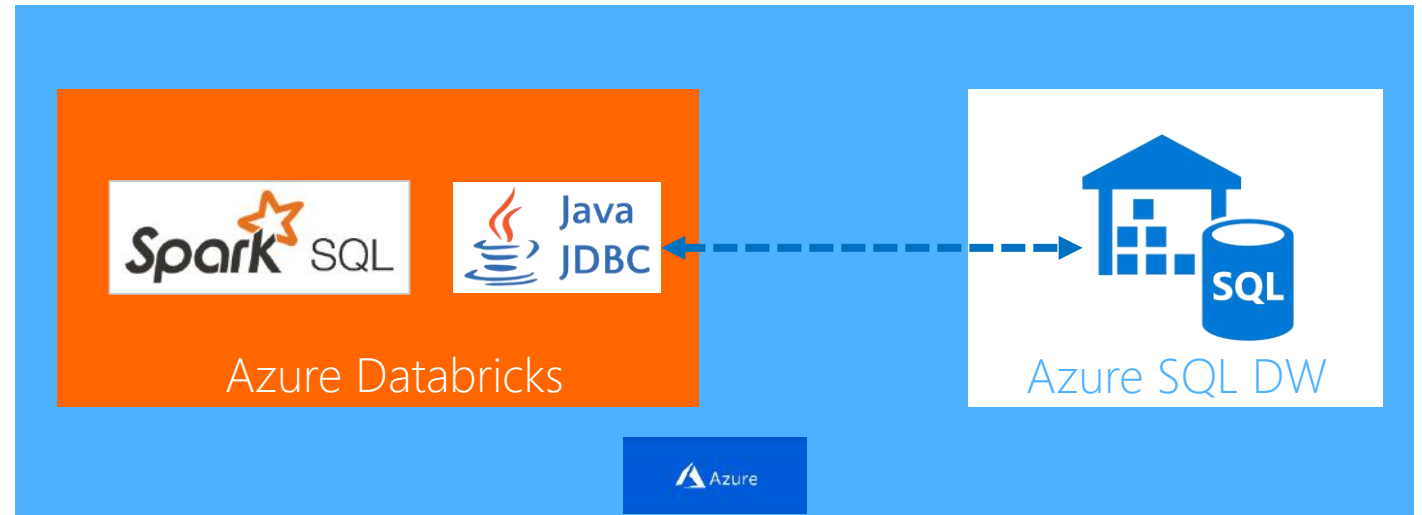
POWER BI INTEGRATION

Enables powerful visualization of data in Spark with Power BI

 Power BI is a business analytics tool that provides data Visualization, Report and Dashboard throughout an organization

Power BI Desktop can connect to Azure Databricks clusters to query data using JDBC/ODBC server that runs on the driver node.

- This server listens on port 10000 and it is not accessible outside the subnet where the cluster is running.

- Azure Databricks uses a public HTTPS gateway

- The JDBC/ODBC connection information can be obtained from the Cluster UI directly as shown in the figure.

- When establishing the connection, you can use a Personal Access Token to authenticate to the cluster gateway. Only users who have attach permissions can access the cluster via the JDBC/ ODBC endpoint.

- In Power BI desktop you can setup the connection by choosing the ODBC data source in the "Get Data" option.

Databricks Delta

Structured streaming ninja

Announcing Databricks Delta

First **UNIFIED** data management system that delivers:

The
SCALE
of data lake

The
**RELIABILITY &
PERFORMANCE**
of data warehouse

The
LOW-LATENCY
of streaming

databricks

Databricks Delta Under the Hood

MASSIVE SCALE

- Decouple Compute & Storage

RELIABILITY

- ACID Transactions & Data Validation
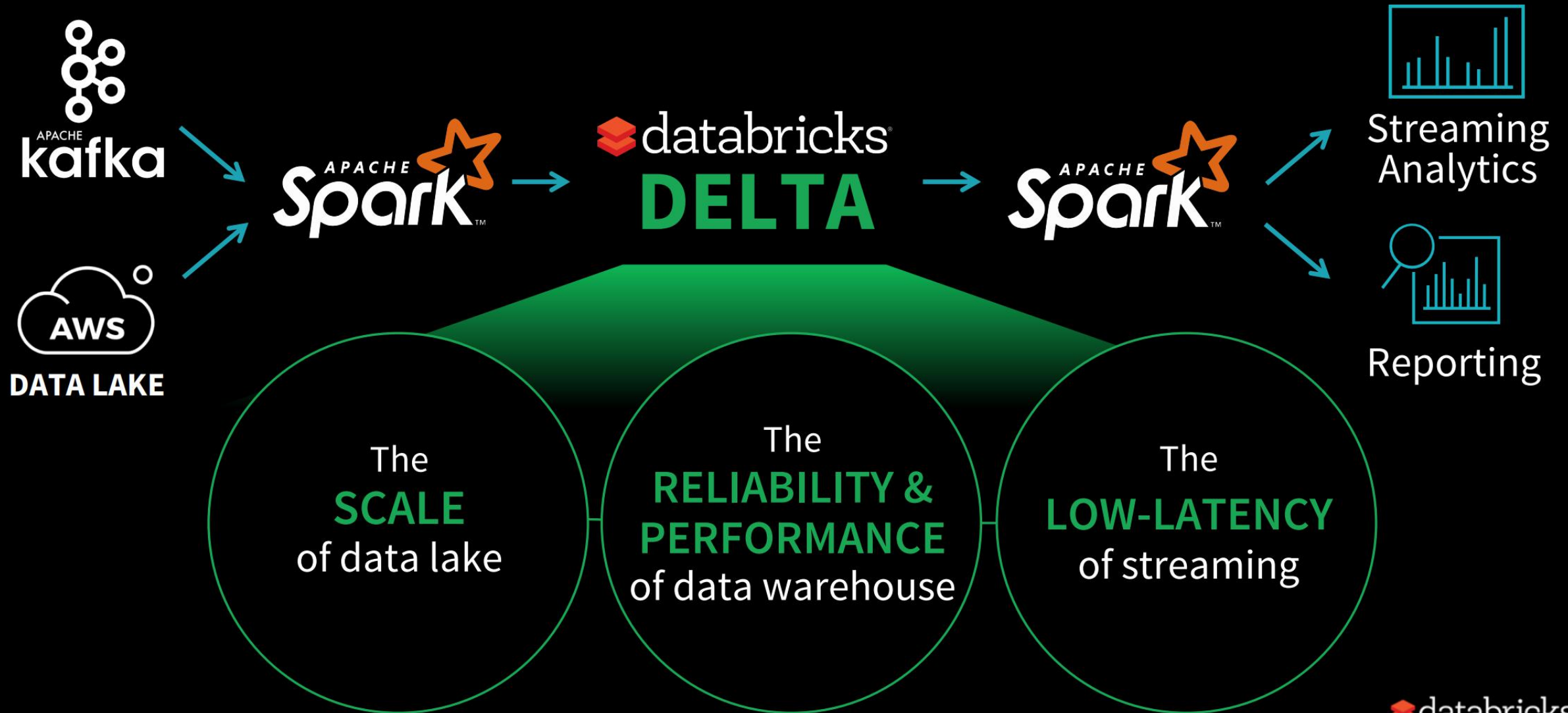
PERFORMANCE

- Data Indexing & Caching (10-100x)

LOW-LATENCY

- Real-Time Streaming Ingest

databricks

Query Sample

```
events =
spark.readStream.json("/data/events")
events.writeStream
    .format("delta")
    .outputMode("append")
    .option("checkpointLocation",
"/delta/events/_checkpoint/etl-from-json")
    .start("/delta/events")
```

The Delta Architecture

Case studies

Two customer stories

Microsoft Azure Databricks

Microsoft

"Every day, we analyze nearly a terabyte of wind turbine data to optimize our data models. Before, that took several hours. With Microsoft Azure Databricks, it takes a few minutes. This opens a whole range of possible new applications."

Sam Julian, Product Owner, Data Services, E.ON

e·on

Based in Germany, E.ON is an international, privately owned energy supplier focusing on renewables and energy networks.

Microsoft Azure Databricks

"At Lennox International, we have 1000's of devices streaming data back into our IoT environment. With Azure Databricks, we moved from 60% accuracy to 94% accuracy on detecting equipment failures. Using Azure Databricks has opened the flood gates to all kinds of new use cases and innovations. In our previous process, 15 devices, which created 2 million records, took 6 hours to process. With Azure Databricks, we are able to process 25,000 devices – 10 billion records – in under 14 minutes."

Sunil Bondalapati, Director of IT, Lennox International

Lennox International is a global leader in the heating, air conditioning, and refrigeration markets.

Вадим Челышков

vachel@microsoft.com

Cloud Solution Architect – Data & AI

Enterprise Commercial, Microsoft Russia
Mobile: +7 966-071-79-00
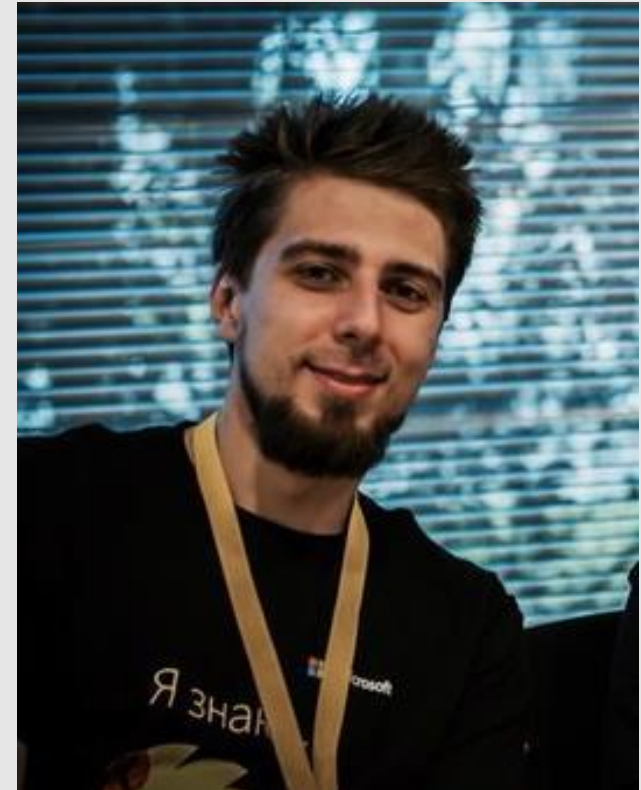
Microsoft

Вопросы/Предложения?

Ждем вас в отдельной переговорной комнате за зоной ресепшена (#1).

Ее можно найти по надписи "Microsoft".