FUTURE DEV DAY

ORGANIZATOR Future Processing

2018 1 GRUDNIA

Praktyczne wykorzystanie architektury Lambda do przetwarzania Big Data na platformie Azure

Tomasz Krawczyk

tkrawczyk@future-processing.com

WWW.FUTUREDEVDAY.PL

Agenda

- Big Data
- Lambda Architecture
- Big Data Project
- Azure as a Big Data Platform
- Our Solution

Big Data 3V

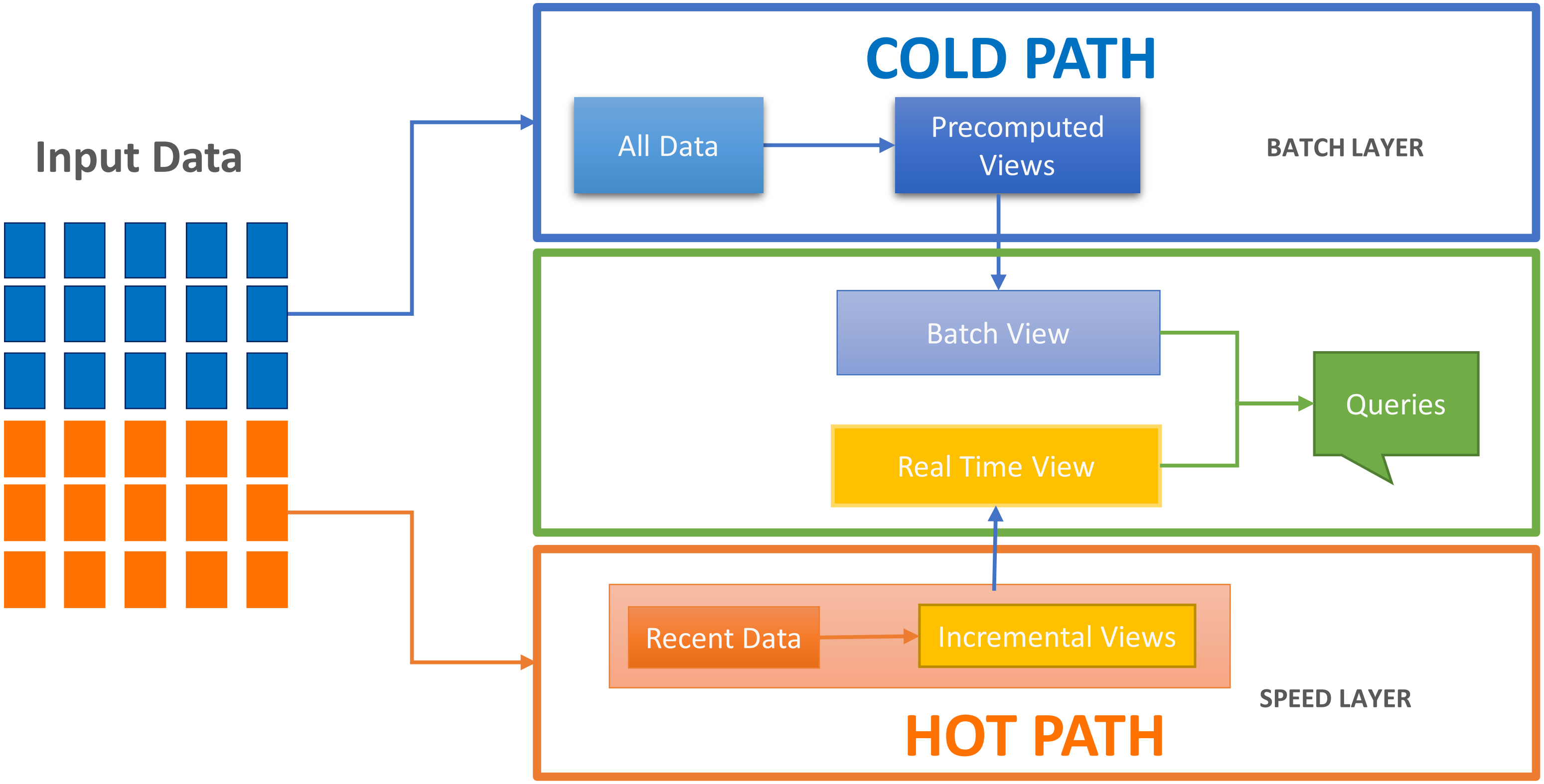
- Data Volume

Byte	One grain of rice
Kilobyte	Cup of rice
Megabyte	8 bags of rice
Gigabyte	3 semi trucks
Terabyte	2 container ships
Petabyte	Blankets Manhattan
Exabyte	Blankets west coast states
Zettabyte	Fills the Pacific Ocean
Yottabyte	As earth-sized rice ball

- Data Variety
  - Structured
  - Unstructured
  - Semi-structured
  - All the above

- Data Velocity
  - Near to Real Time
  - Batch

Lambda Architecture



Input Data

COLD PATH

All Data

Precomputed Views

BATCH LAYER

Batch View

Real Time View

Queries

Recent Data

Incremental Views

SPEED LAYER

HOT PATH

Data Lake Approach

What is Data Lake ?

“If you think of a **datamart** (a subset of a data warehouse) as a store of bottled water – cleansed and packaged and structured for easy consumption – the **data lake** is a large body of water in a more **natural state**„

Pentaho CTO James Dixon

Source: https://premiumwaters.com

Source :https://snowbrains.com

I(ngest) S(tore) A(nalyse) S(urface) A(ct)

Make Me More Money

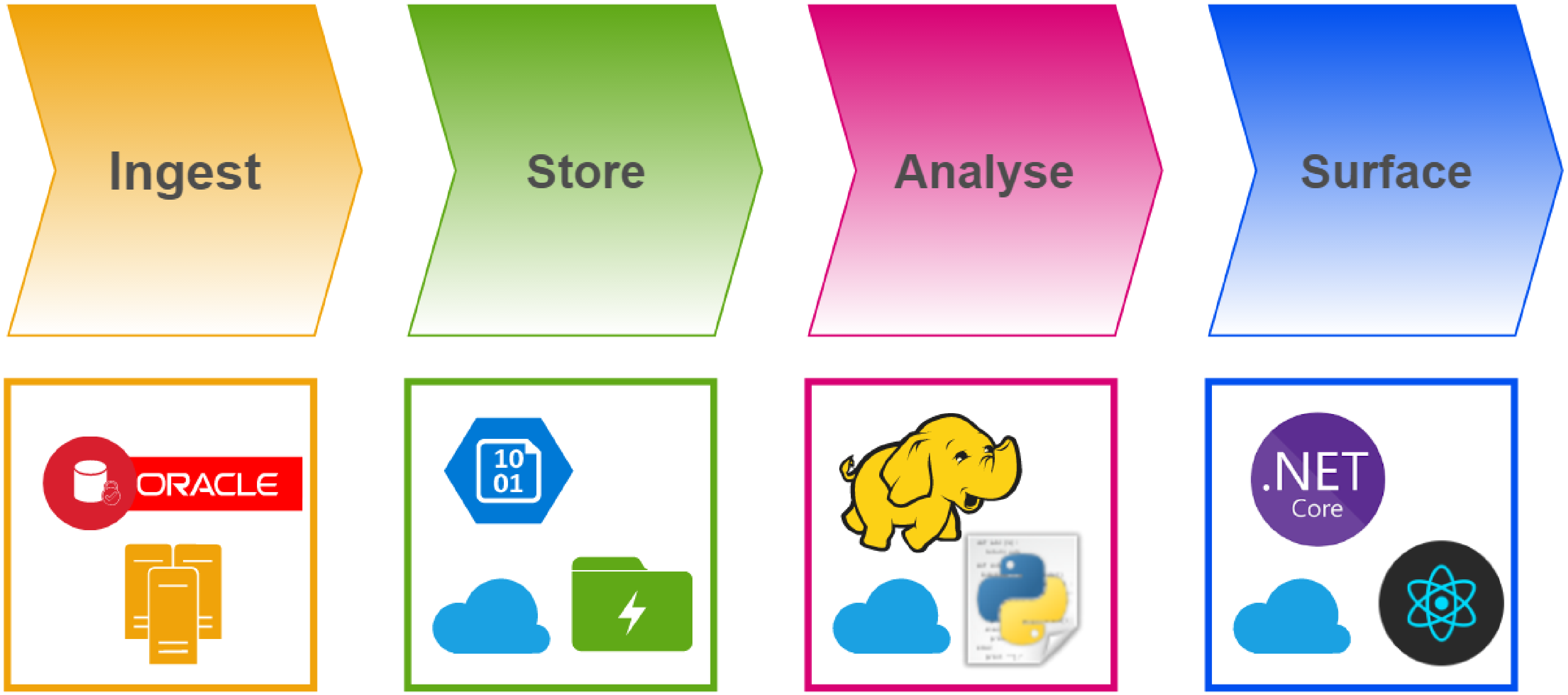
Big Data Project

- **Input Data**
  - IoT (400 000 Meters)
  - Source OnPremise Oracle Database
  - 30 TB Initial Load
  - 15 GB Daily Load (Batch Mode)

- **Output data**
  - KPIs
  - Visualizations (Maps, Charts…)
  - Access to raw data
    - Detailed Queries (Point Queries)

- **Data Processing**
  - 7 problems = 7 algorithms (Mathematical and analytical models)
  - Batch mode
  - Total Processing Time < 8h

Big Data Project – Basic Concept



Ingest

Store

Analyse

Surface

OnPremise + Oracle

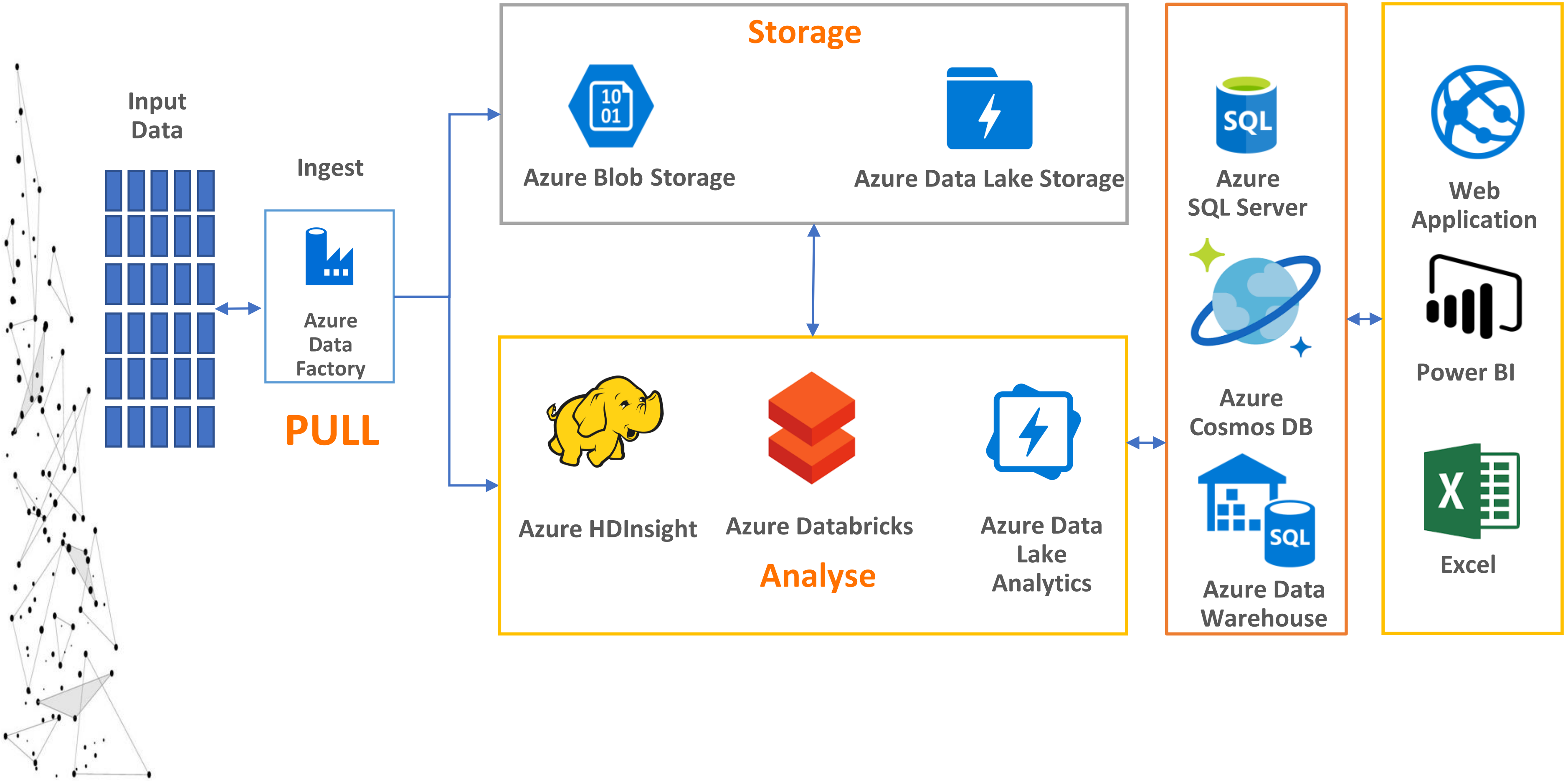
Azure Storage

Azure + Hadoop + Python

Azure + .Net Core + React

FP Data Solutions

Azure – Lambda architecture (Cold Path)



Azure – Big Data Storage

Azure Blob Storage

- General purpose object store
- Object store with flat namespace
- Hot/cold/archive tiers
- Data replication and redundancy options

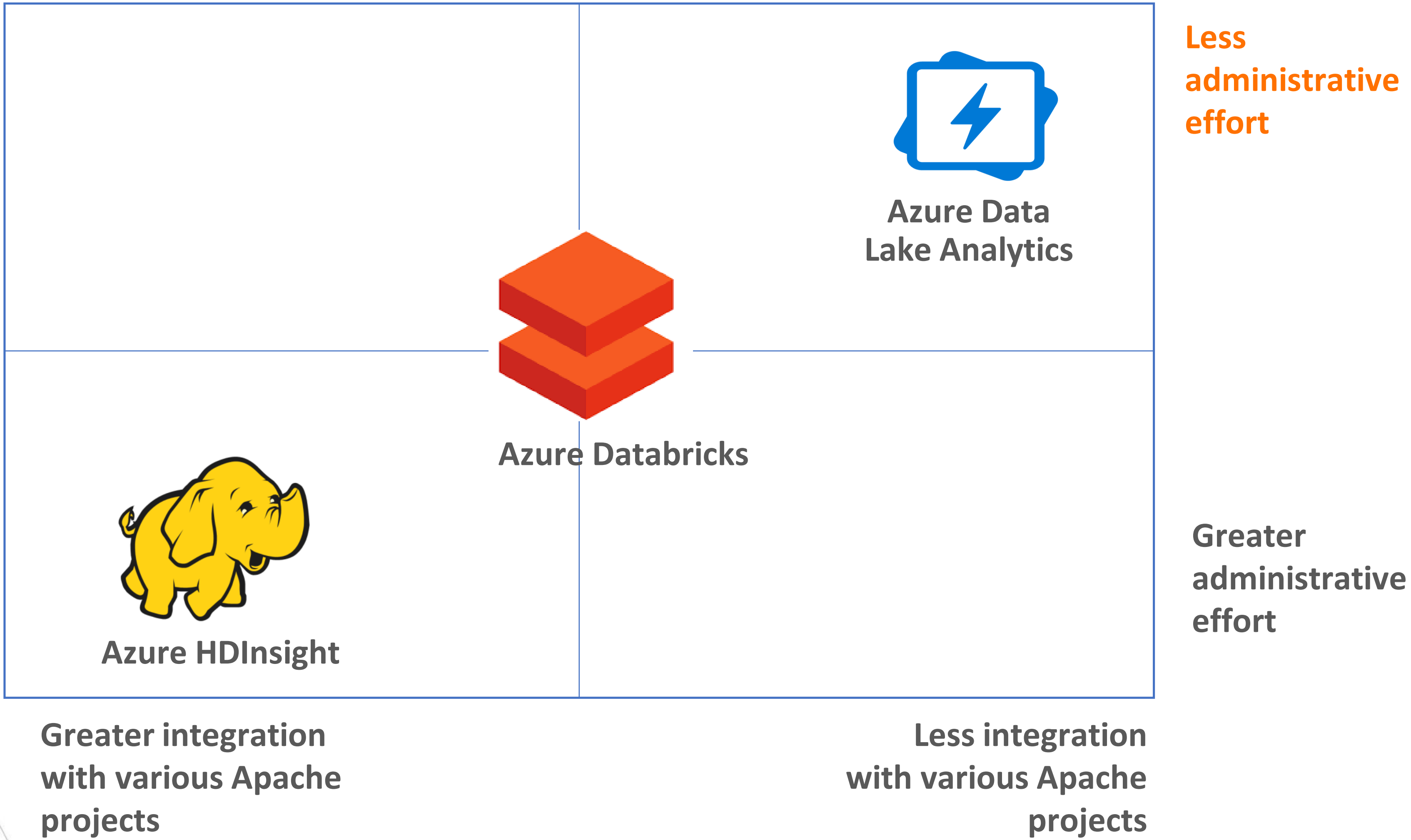
Azure Data Lake Storage (Gen1)

- Unlimited storage, petabyte files
- **WebHDFS**-compatible REST interface
- Hadoop and big data optimizations
- Supports files and folders objects

Azure Data Lake Storage (Gen2)

- Multi-modal combining features from both of the above
- Not a separate service: Azure Storage with new features

Azure Big Data - Compute



Less administrative effort

Greater administrative effort

Greater integration with various Apache projects

Less integration with various Apache projects

Azure Data Lake Analytics

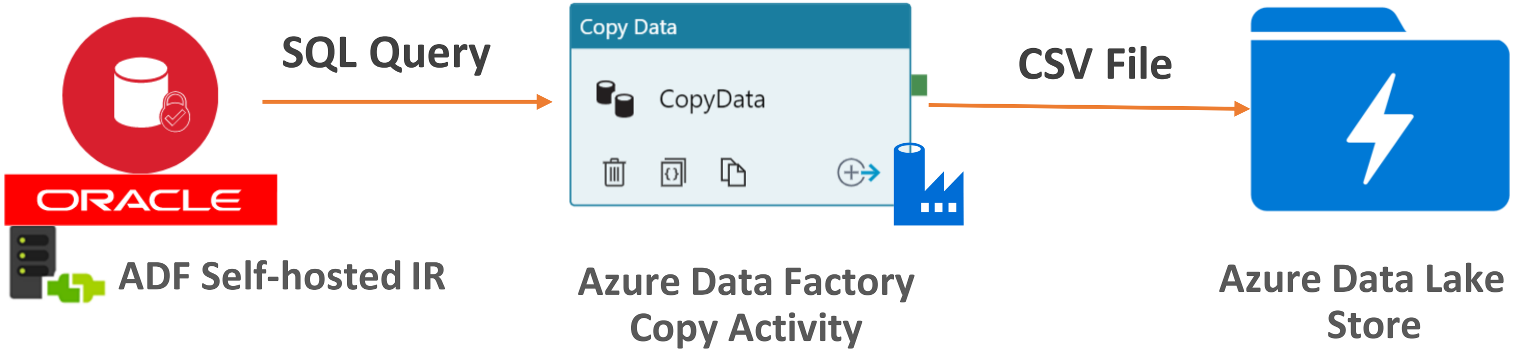
Azure Databricks

Azure HDInsight

Azure Data Factory

- Fully managed service to support **orchestration of data movement and transformation**
- Connect to relational or non-relational data that is **on-premises** or in the **cloud**
- **Allows monitor and manage data processing pipelines**
- Version 1 and **2** (+SSIS)

Loading Data - Ingest



SQL Query

ADF Self-hosted IR

Azure Data Factory Copy Activity

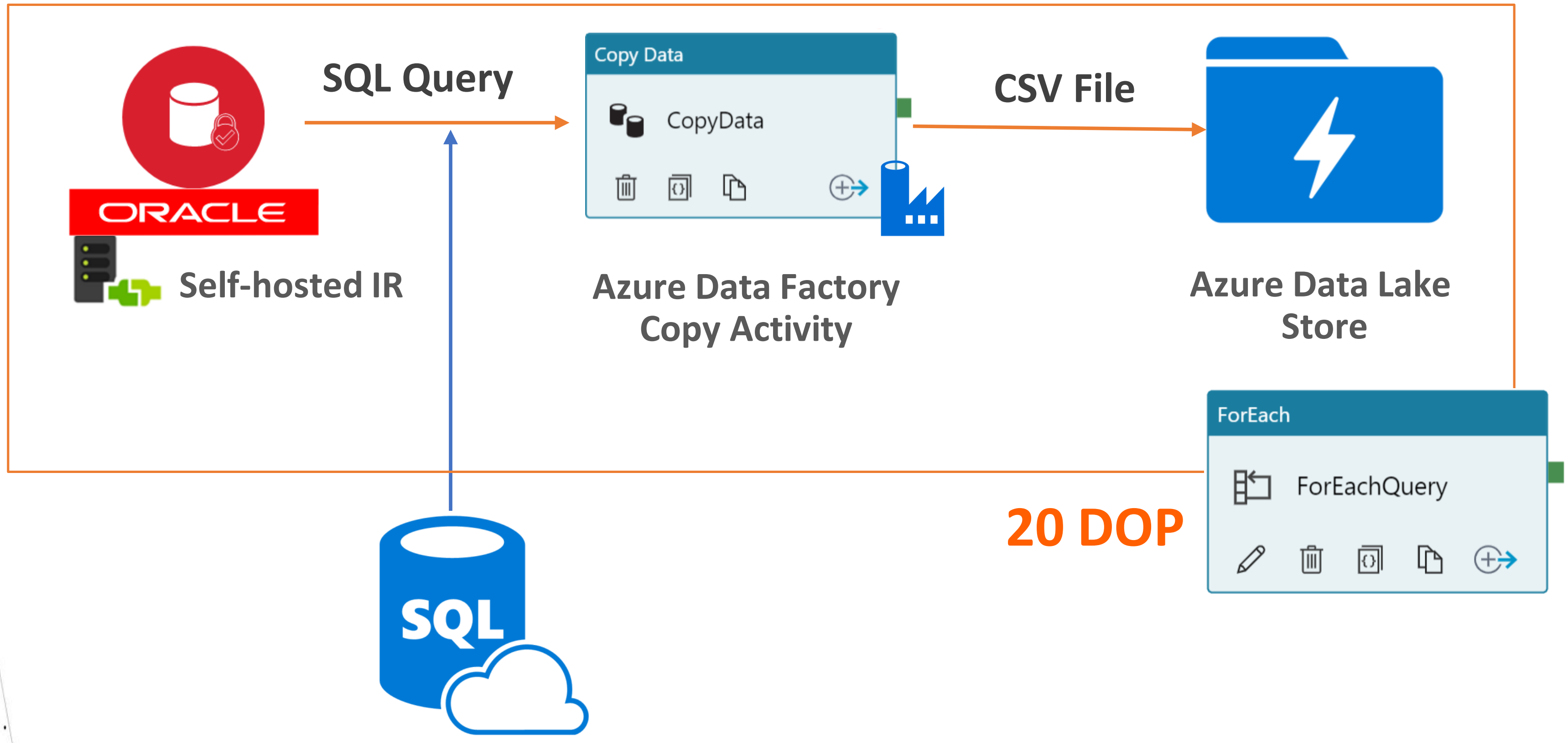
CSV File

Azure Data Lake Store

Challenges

- More than 100 Queries
- Incremental Load

Loading Data - Ingest



SQL Query

Copy Data

CopyData

CSV File

ORACLE

Self-hosted IR

Azure Data Factory Copy Activity

Azure Data Lake Store

ForEach

ForEachQuery

20 DOP

SQL

Query = Where (**Inserted Date** Between **Last Load** and **Now**)

Data Processing - Analyse

[Azure HDInsight

OR

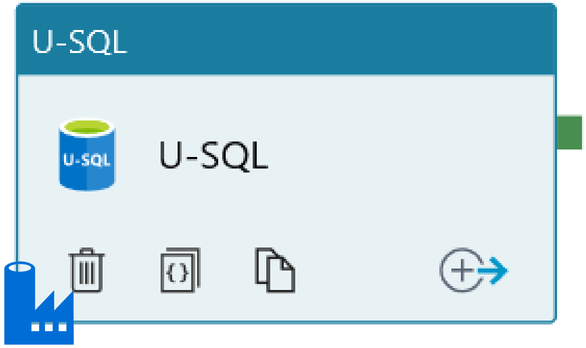
Azure Databricks]

APACHE Spark™

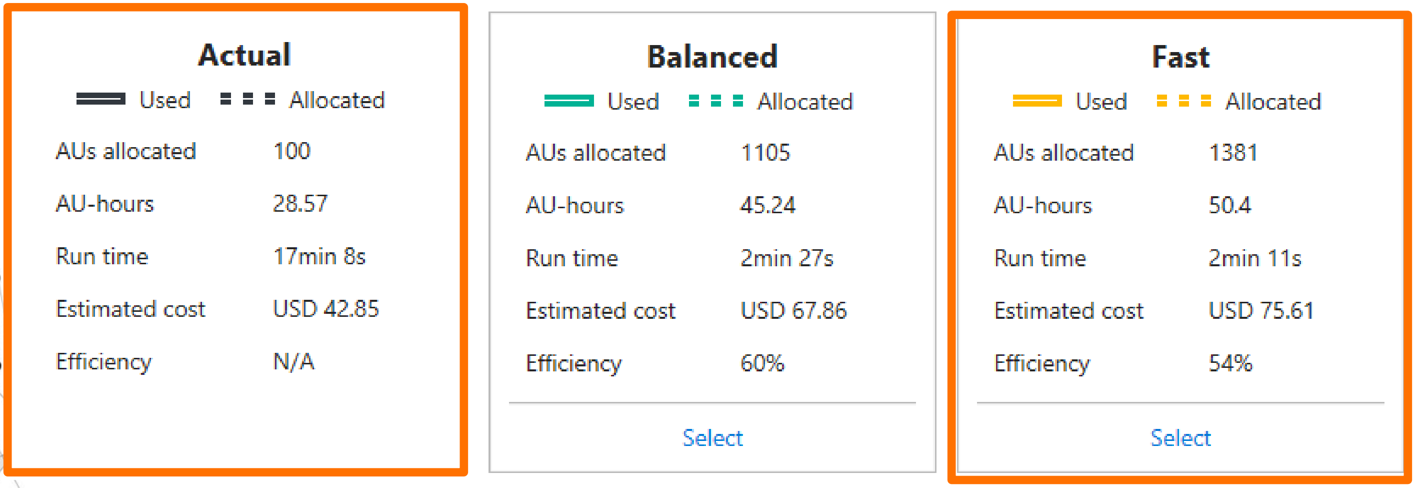
AND

Azure Data Lake Analytics

Data Processing – Basic Analysis



ADLUs = 100



Actual

▬ Used ▪▪▪ Allocated

AUs allocated	100
AU-hours	28.57
Run time	17min 8s
Estimated cost	USD 42.85
Efficiency	N/A

Balanced

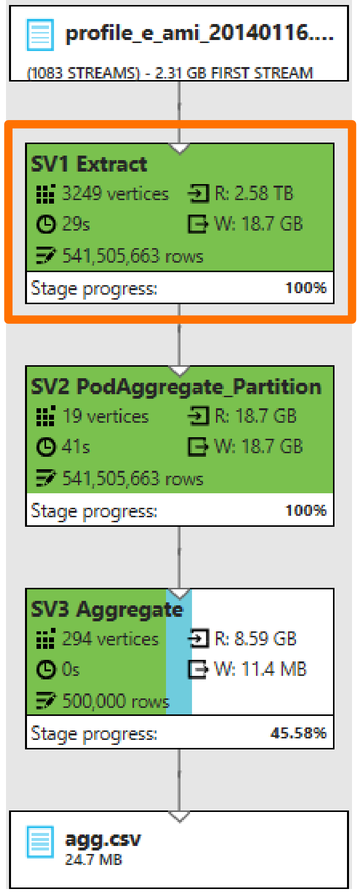
▬ Used ▪▪▪ Allocated

AUs allocated	1105
AU-hours	45.24
Run time	2min 27s
Estimated cost	USD 67.86
Efficiency	60%

Fast

▬ Used ▪▪▪ Allocated

AUs allocated	1381
AU-hours	50.4
Run time	2min 11s
Estimated cost	USD 75.61
Efficiency	54%



profile_e_ami_20140116....
(1083 STREAMS) - 2.31 GB FIRST STREAM

SV1 Extract
3249 vertices R: 2.58 TB
29s W: 18.7 GB
541,505,663 rows
Stage progress: 100%

SV2 PodAggregate_Partition
19 vertices R: 18.7 GB
41s W: 18.7 GB
541,505,663 rows
Stage progress: 100%

SV3 Aggregate
294 vertices R: 8.59 GB
0s W: 11.4 MB
500,000 rows
Stage progress: 45.58%

agg.csv
24.7 MB

Azure Data Lake Analytics

Data Processing - Advanced Analysis

Why APACHE Spark™

python™

Python is a king of data science

Data sources and sinks

APACHE HBASE

SQL

HIVE

Cassandra
A Highly Scalable NoSQL Database

APACHE PHOENIX

Microsoft SQL Server

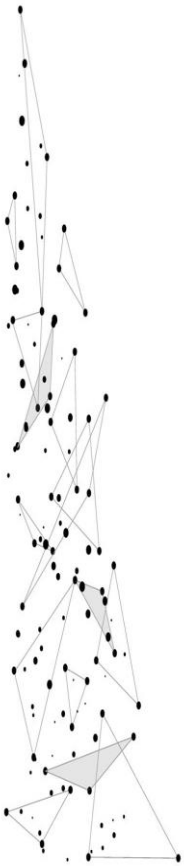
SQL

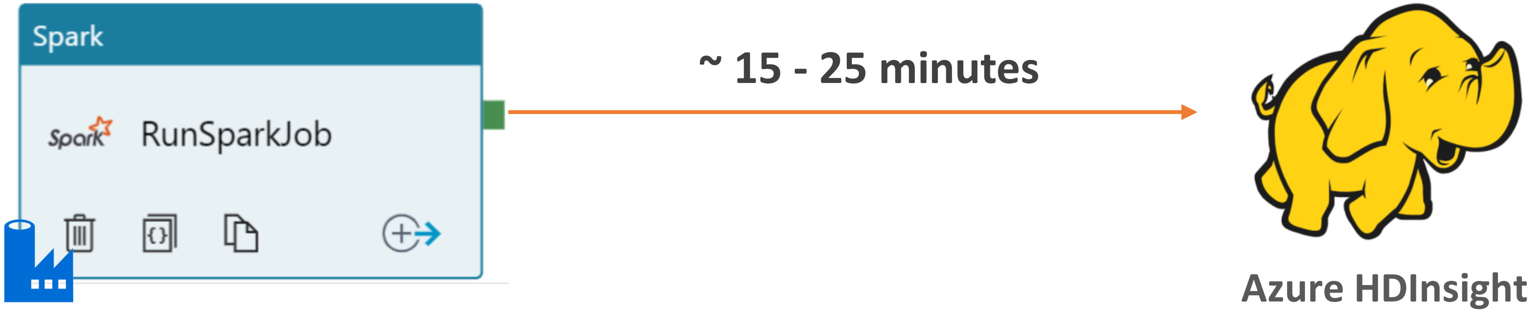
PostgreSQL

10
01

ORACLE

Data Processing - Analyse





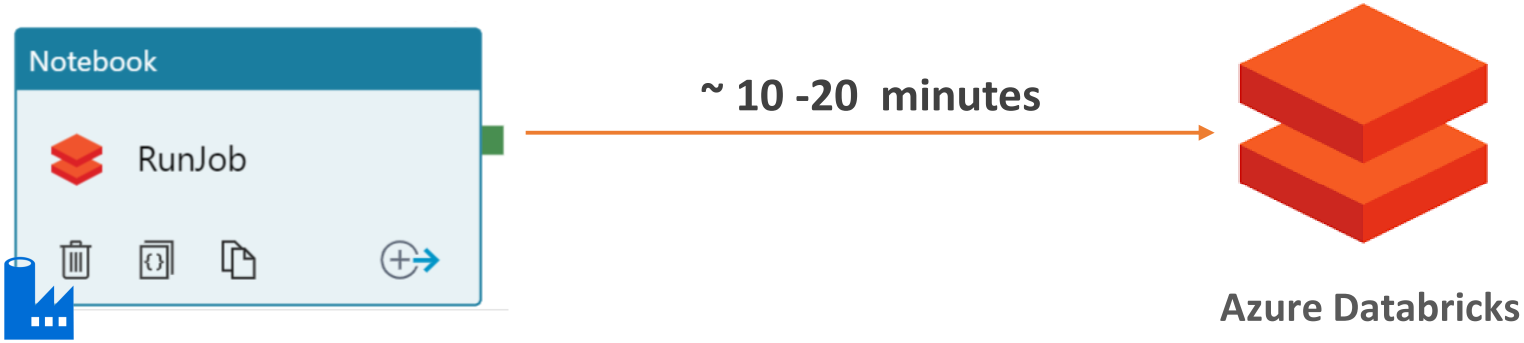
Spark

RunSparkJob

~ 15 - 25 minutes

Azure HDInsight

Create Cluster on demand, run job and terminate cluster



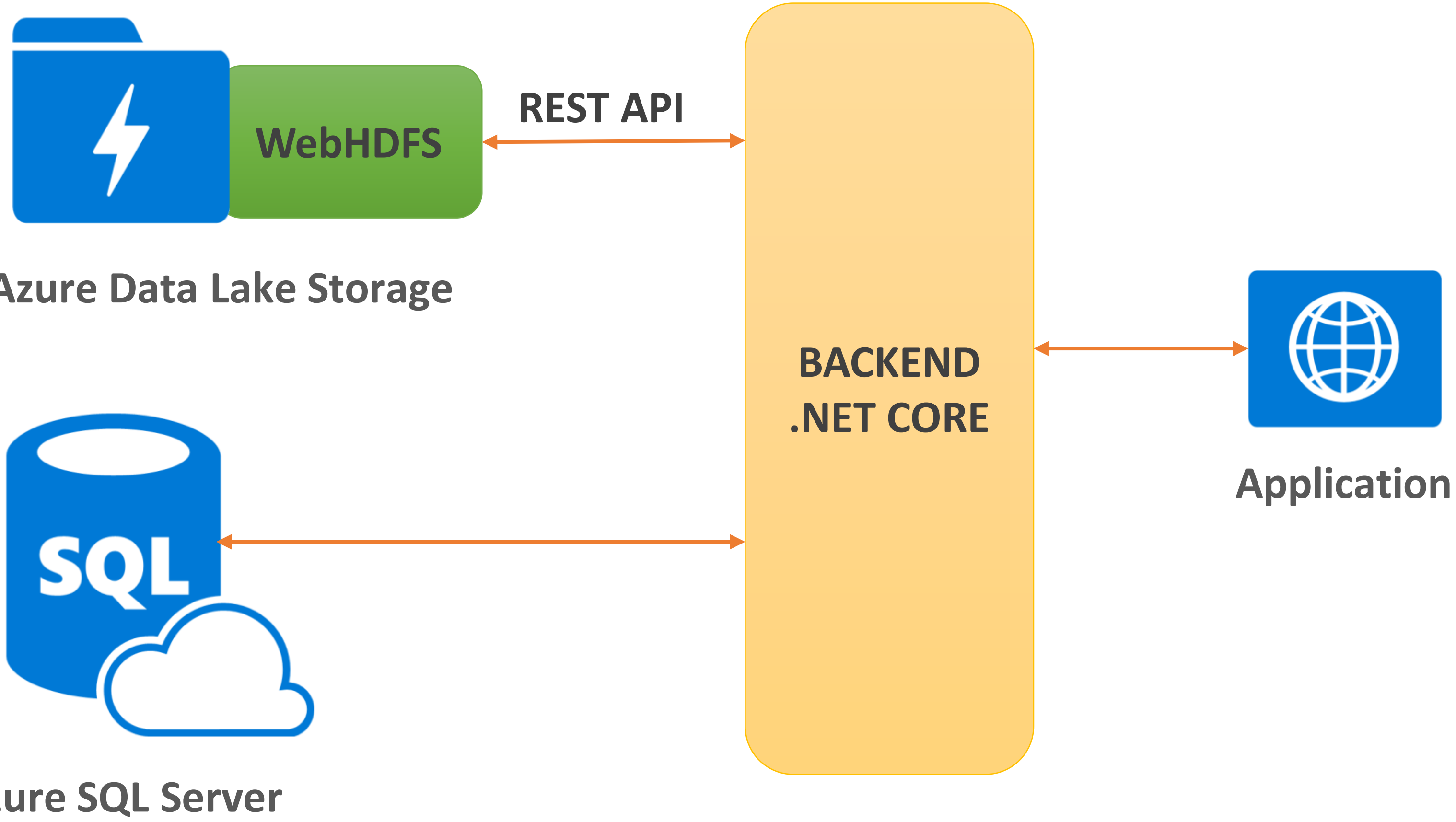
Notebook

RunJob

~ 10 -20 minutes

Azure Databricks

Data Processing – Results –Interactive Queries



Results - Interactive Queries

Query:

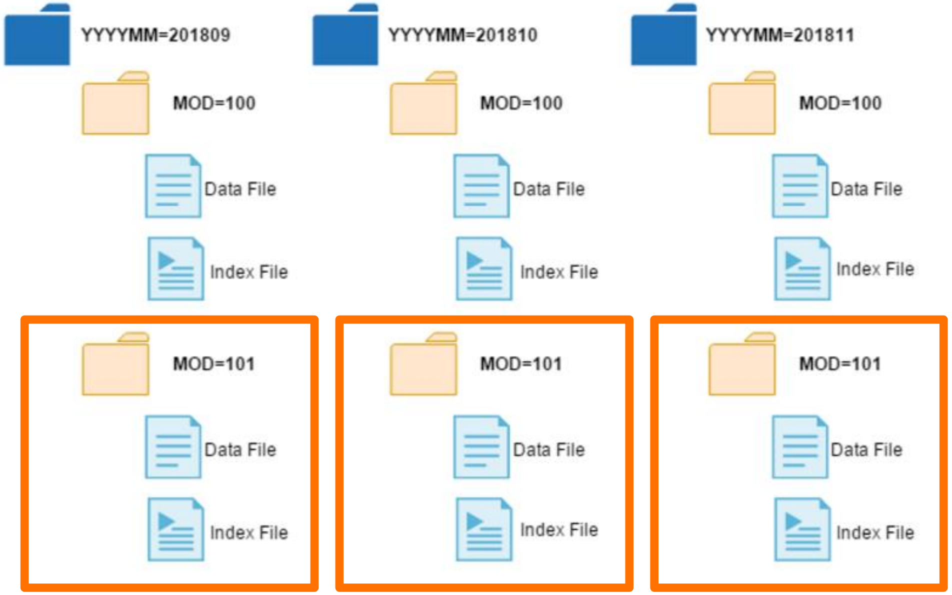
ObjectId = 1101 Date Between 2018-09-01 and 2018-11-30

MOD = 1101 % 1000 = 101

YYYYMM = 201809

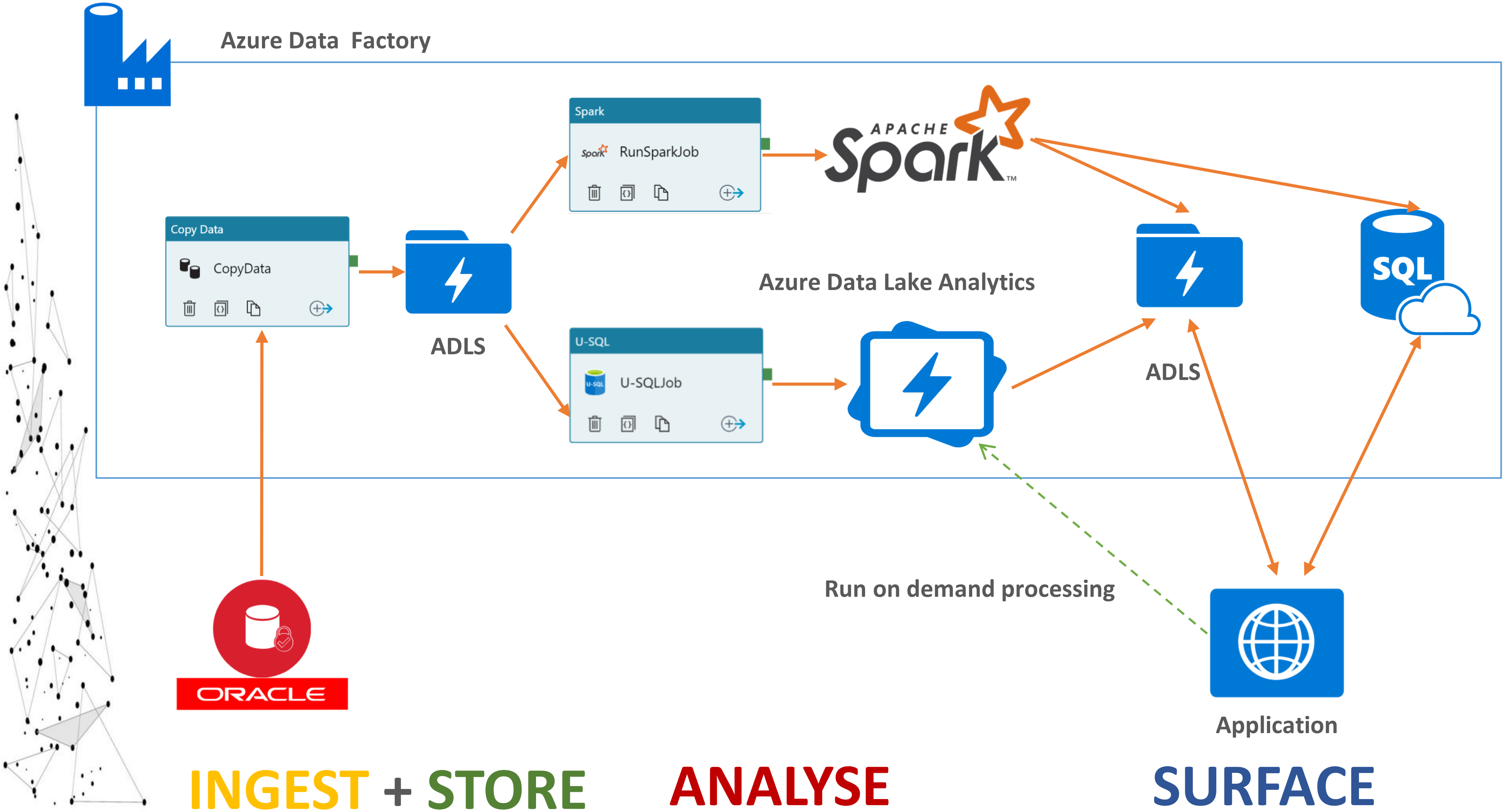
YYYYMM = 201810

YYYMM = 201811



Result = (Read Part1 (YYYYMM=201810 MOD =101) + Read Part2 (YYYYMM=201811 MOD =101) + Read Part3 (YYYYMM=201811 MOD =101))+ Merge

System Architecture



INGEST + STORE ANALYSE SURFACE

THANK YOU!

tkrawczyk@future-processing.com