Analyze Billions of Rows of Data in Real-Time Using Azure Data Explorer

niels berglund

Software Architect Lead - Derivco
niels.it.berglund@gmail.com
https://nielsberglund.com
https://linkedin.com/in/nielsberglund
@nielsberglund

DERIVCO

MVP Microsoft® Most Valuable Professional

Session Feedback

Session Feedback

- Session feedback is important!
- SQLBits donates to National Trust!
- You can WIN prizes!

https://sqlb.it/?7090

Agenda

- Data
- Azure Data Explorer
- Ingestion
- Query data

Top 5 Reasons to Attend SQLBits

Top 5 Reasons to Attend SQLBits

5. Learn new things

4. Get to hear interesting stuff

3. Get away from home

2. Hang out with your brethren

1. Receive SWAG

DERIVCO

Data

Data

It's All About the Base (Meghan Trainor 2014)

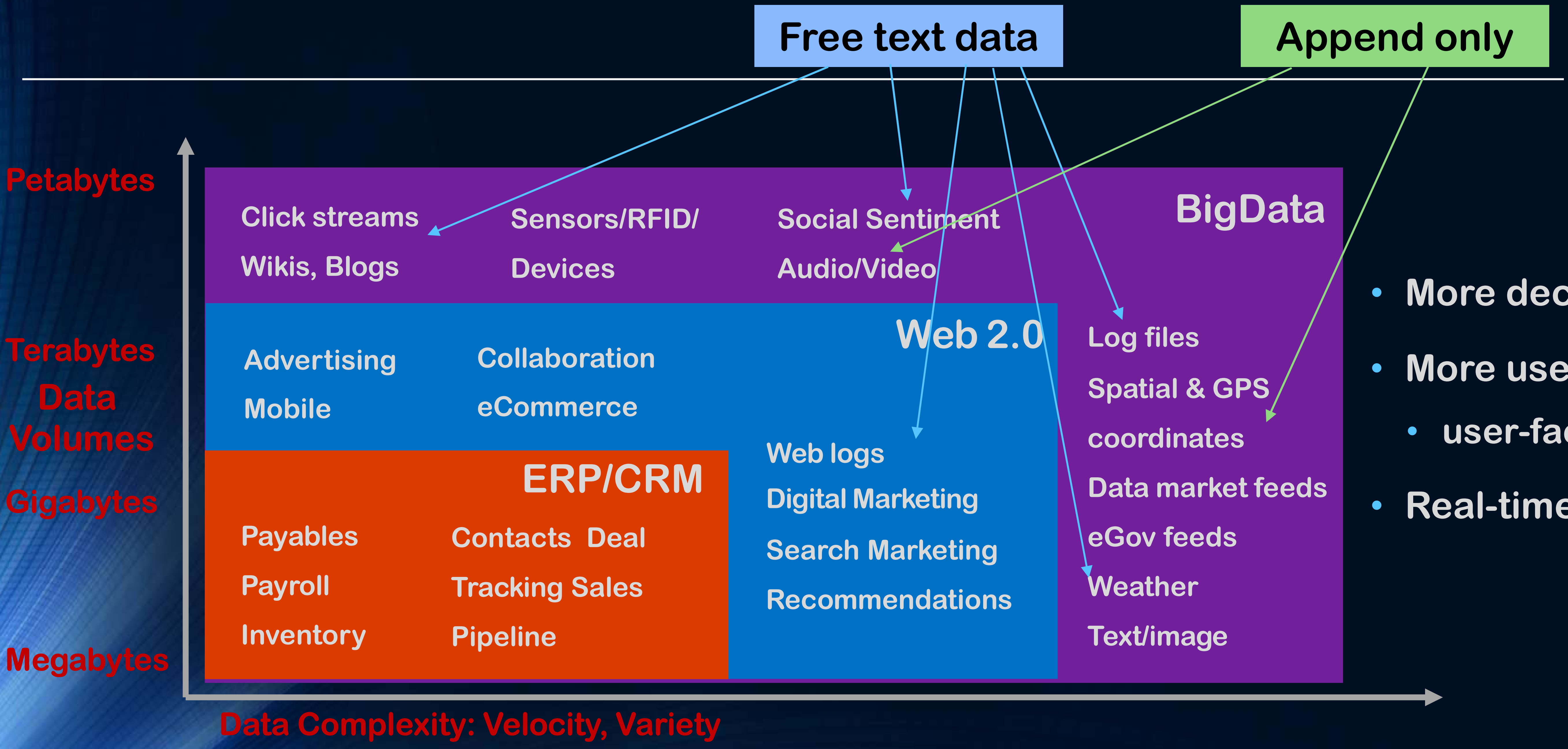
It's All About the Data

- We generate more and more data.
  - 2020 - 44 ZBs
  - 2025 - 175 ZBs

- While data grows 400% …
- … **less than 30% gets analyzed! ☹**

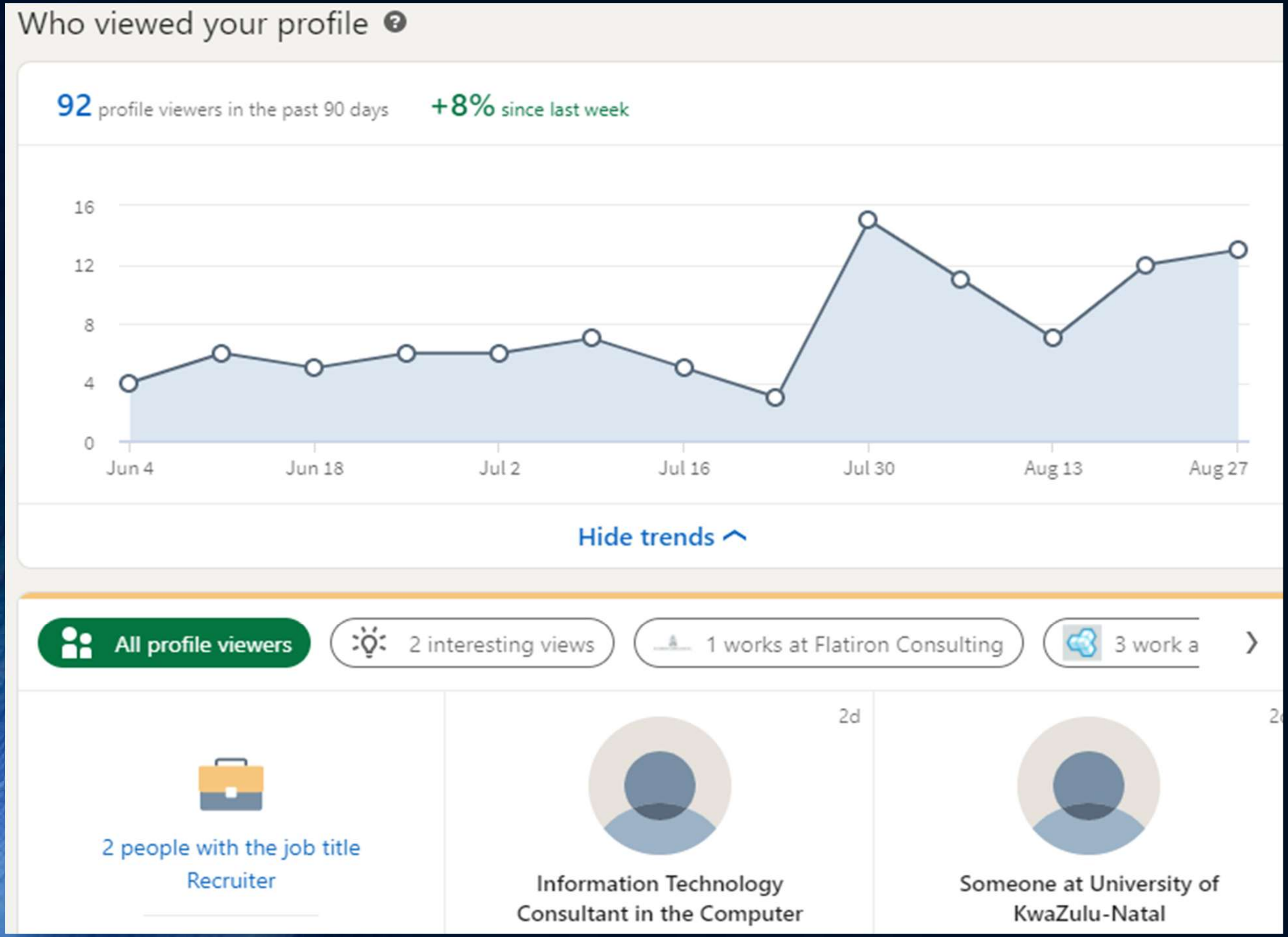
Big Data & Modern Business

Big Data & Modern Business



- More decision makers
- More users
  - user-facing
- Real-time decisions

User Facing Analytics



Who viewed your profile

92 profile viewers in the past 90 days +8% since last week

Hide trends

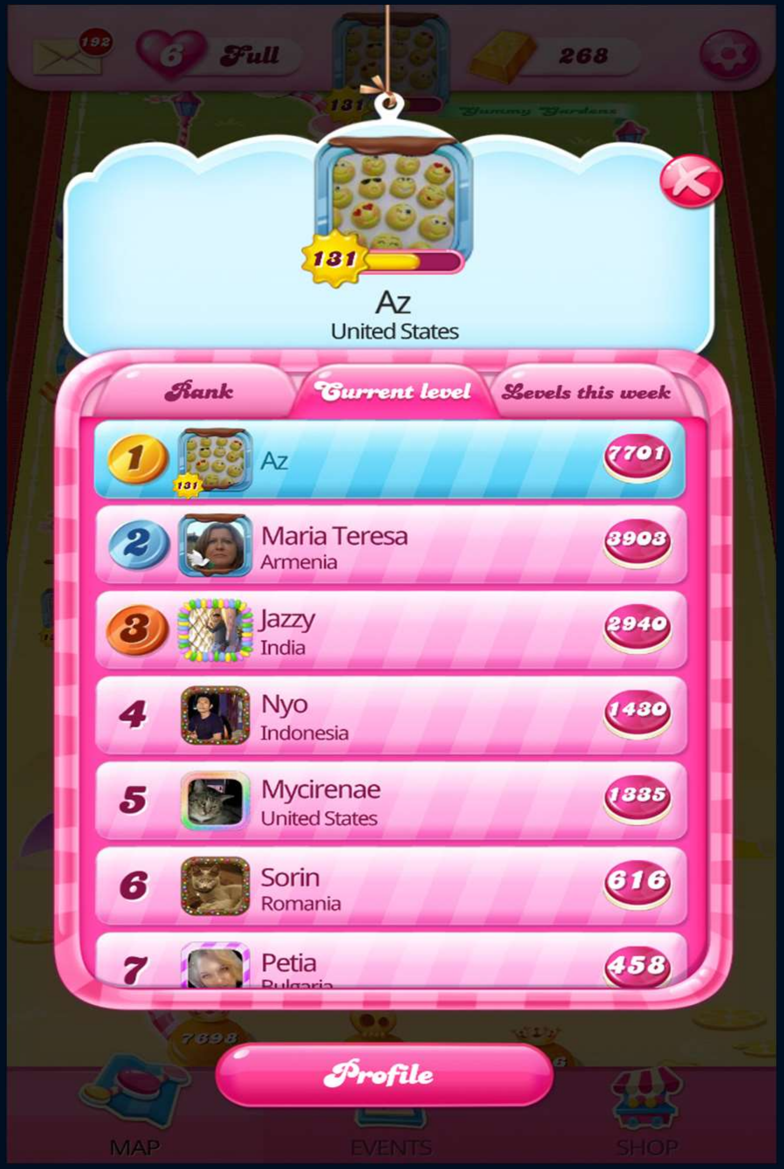
2 people with the job title Recruiter

2d

Information Technology Consultant in the Computer

2

Someone at University of KwaZulu-Natal



Az
United States

1	Az	7701
2	Maria Teresa Armenia	3903
3	Jazzy India	2940
4	Nyo Indonesia	1430
5	Mycirenae United States	1335
6	Sorin Romania	616
7	Petia Bulgaria	458

Azure Data Explorer

Azure Data Explorer

- Fully managed Big Data Analytics platform.
- High performance.
- Analyzes high volumes of data in near real-time.
- End-to-end solution for data ingestion.
- Query, visualization, and management.
- Useful for log analytics, time series, IoT, and general-purpose exploratory.

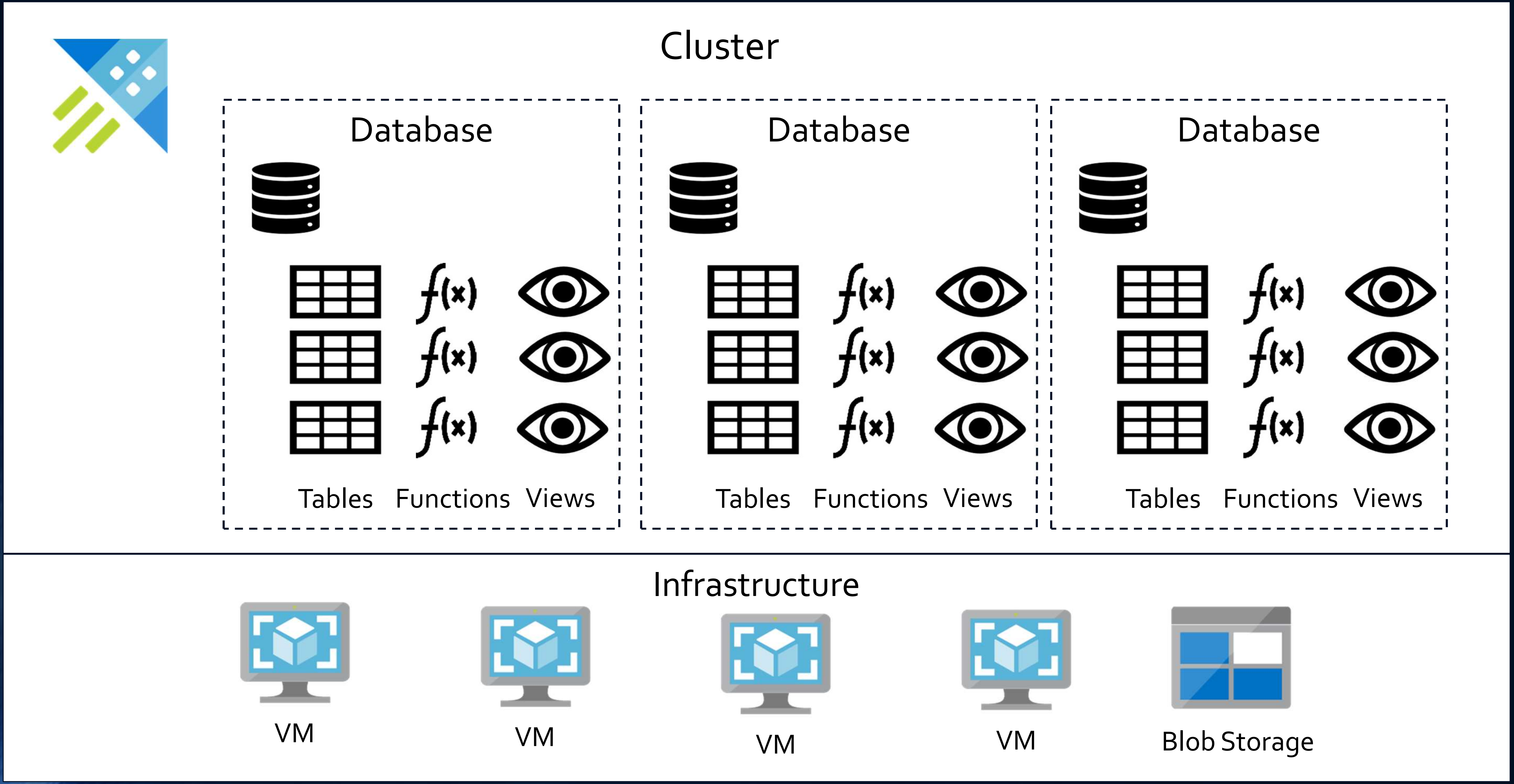
Azure Data Explorer - II

Azure Data Explorer - II

- Ability to work with any kind of data: structured, semi-structured (JSON and more) and unstructured (free text).
- User friendly query language.
- Advanced analytics.
- Versatile data visualization.
- Automatic ingest, process and export

Architecture

ADX: Architecture - I

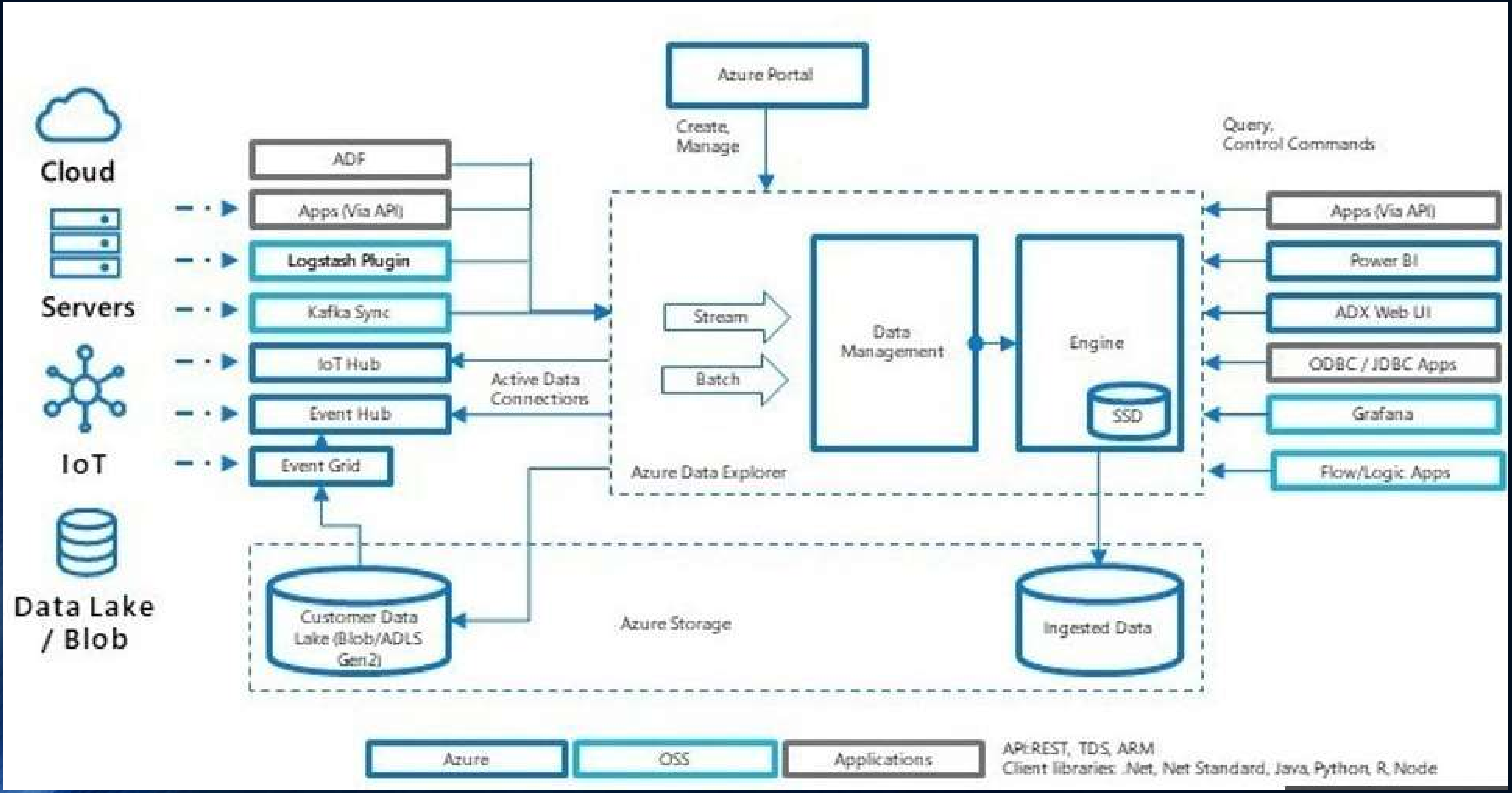


ADX: Architecture - II

- Two main services in ADX:
  - Engine Service
  - Data Management Service

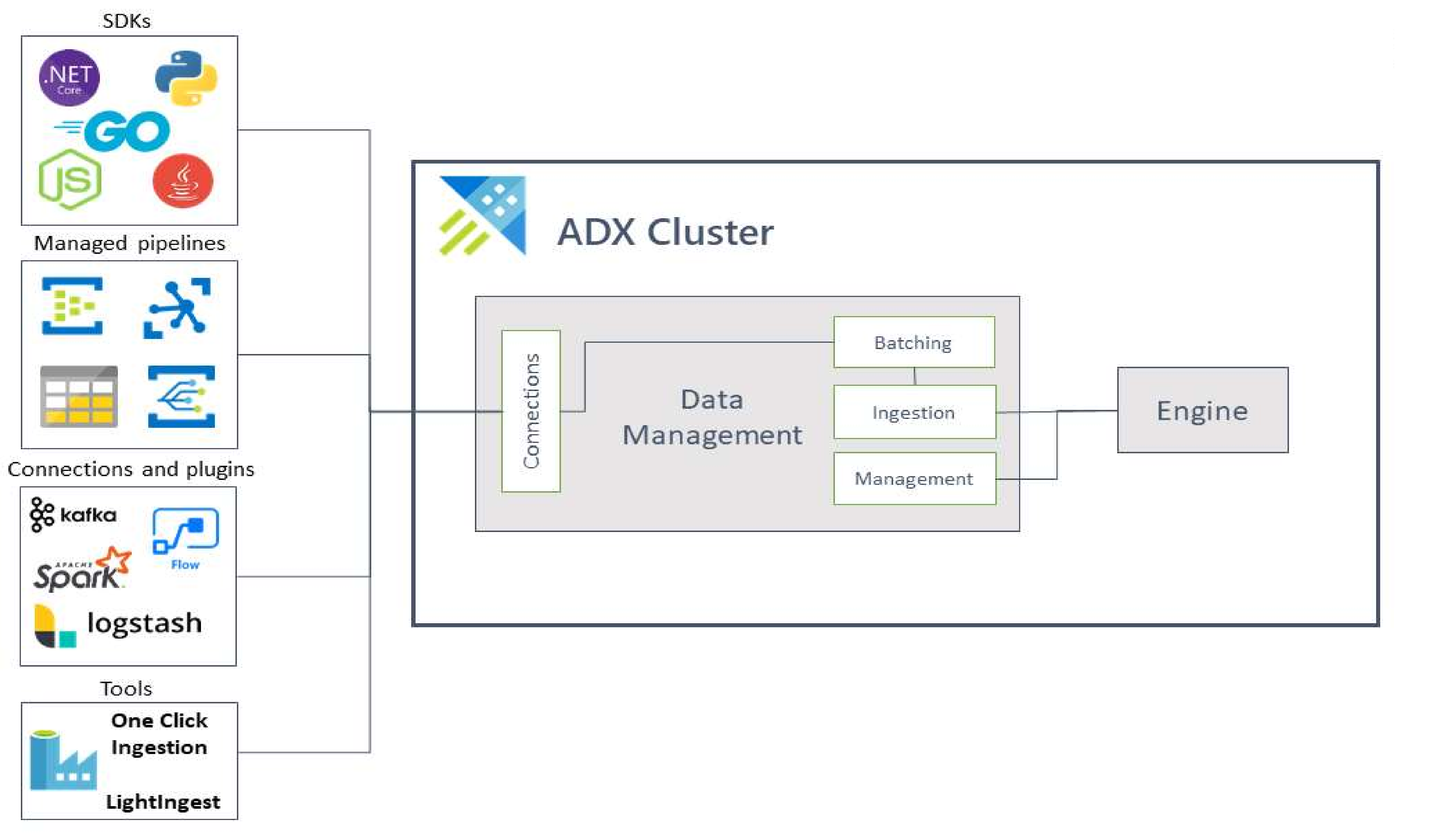
- Engine service:
  - responsible for processing the incoming raw data and serving user queries via an API.
- Data management service:
  - connecting the Engine to the various data pipelines.
  - orchestrating and maintaining continuous data ingestion process from these pipelines
  - data grooming

ADX: Architecture - III



Ingestion

Ingestion Architecture



Ingestion

- Create table
- Set retention policy
  - database or table level.
- The table need to be aware of what data is ingested
  - Ingestion mapping
- Ingestion policy for batch/streaming ingestion

Set-up for Ingestion

```
.create table GamePlay
(PlayerID: int, GameID: int, Win: long, Score: int, EventTime: datetime)
```

```
.create table GamePlay ingestion json mapping 'gameplay_json_mapping'
'[{"column":"PlayerID", "Properties":{"path":"$.playerId"}},
{"column":"GameID", "Properties":{"path":"$.gameId"}},
{"column":"Win", "Properties":{"path":"$.win"}},
{"column":"Score", "Properties":{"path":"$.score"}},
{"column":"EventTime", "Properties":{"path":"$.eventTime"}} ]'
```

```
.alter table ['GamePlay'] policy ingestionbatching
@'{"MaximumBatchingTimeSpan":"00:00:01", "MaximumNumberOfItems": 1,
"MaximumRawDataSizeMB": 300}'
```

```
.alter table ['GamePlay'] policy streamingingestion enable
```

BATCHING

- Optimized for high ingestion throughput
- Preferred method and most performant
- Data is batched according to properties
- Set ingestion batching policy on databases or tables
- Default max batching value is 5 minutes, 1000 items or total of 1 GB
- 4 GB data size limit for a batch ingestion command

STREAMING

- Ongoing data ingestion from a streaming source
- Near real-time latency for small sets of data per table
- Initially ingested to row store
- Then moved to column store extents
- Streaming can be done using ADX client library or supported pipelines/connectors

Query

Querying ADX

- Kusto Query Language - KQL
- Similar to SQL - slightly different syntax
    - uses | to pipe commands
    - equality: ==
- Full text indexing, time series analysis
- Built in machine learning features

Query Samples

```
//count the number of events
GithubEvent
| count

// visualization
GithubEvent
| summarize count() by bin(CreatedAt, 1d)
| render timechart
```

```
// this parses JSON
GithubEvent
| project Actor.display_login
| take 10
```

```
// linear regression
GithubEvent
| where Repo.name in ("Microsoft/vscode", "Microsoft/TypeScript")
| make-series  count() default=0 on CreatedAt in range(datetime(2016-01-01),
    datetime(2019-04-12), 30d) by RepoName = tostring(Repo.name)
| extend (rsquare, slope, variance, rvariance, interception, linefit) =
          series_fit_line(count_)
| project RepoName, CreatedAt, linefit, count_
| render timechart
```

Summary

- We are getting more and more data
  - being able to analyze the data is vital
- Real-time analysis is becoming the norm
  - enabling end-users to do analysis gives a competitive edge
- Azure Data Explorer; big data analytics platform
- KQL query language for ADX

Thank You!

Questions?

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

Session Feedback

https://sqlb.it/?7090