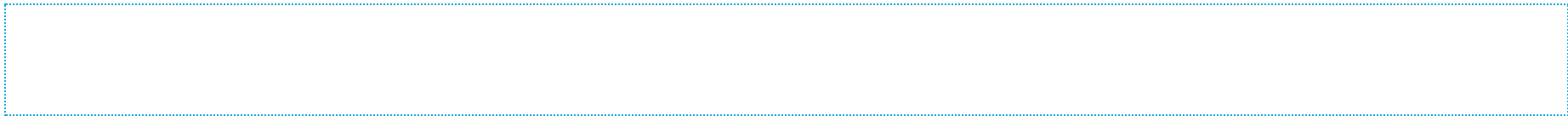


Big Data Analytics

Big data analytics is the use of advanced analytic techniques against very large, diverse big data sets that include structured, semi-structured and unstructured data, from different sources, and in different sizes from terabytes to zettabytes









What is big data exactly?

- Big Data is a collection of data that is huge in volume, yet growing exponentially with time. It is a data with so large size and complexity that none of traditional data management tools can store it or process it efficiently. Big data is also a data but with huge size.
- Sources of data are becoming more complex than those for traditional data because they are being driven by [artificial intelligence \(AI\)](#), mobile devices, social media and the Internet of Things (IoT) .For example, the different types of data originate from sensors, devices, video/audio, networks, log files, transactional applications, web and social media — much of it generated in real time and at a very large scale.

Characteristics of big data

The six Vs of big data

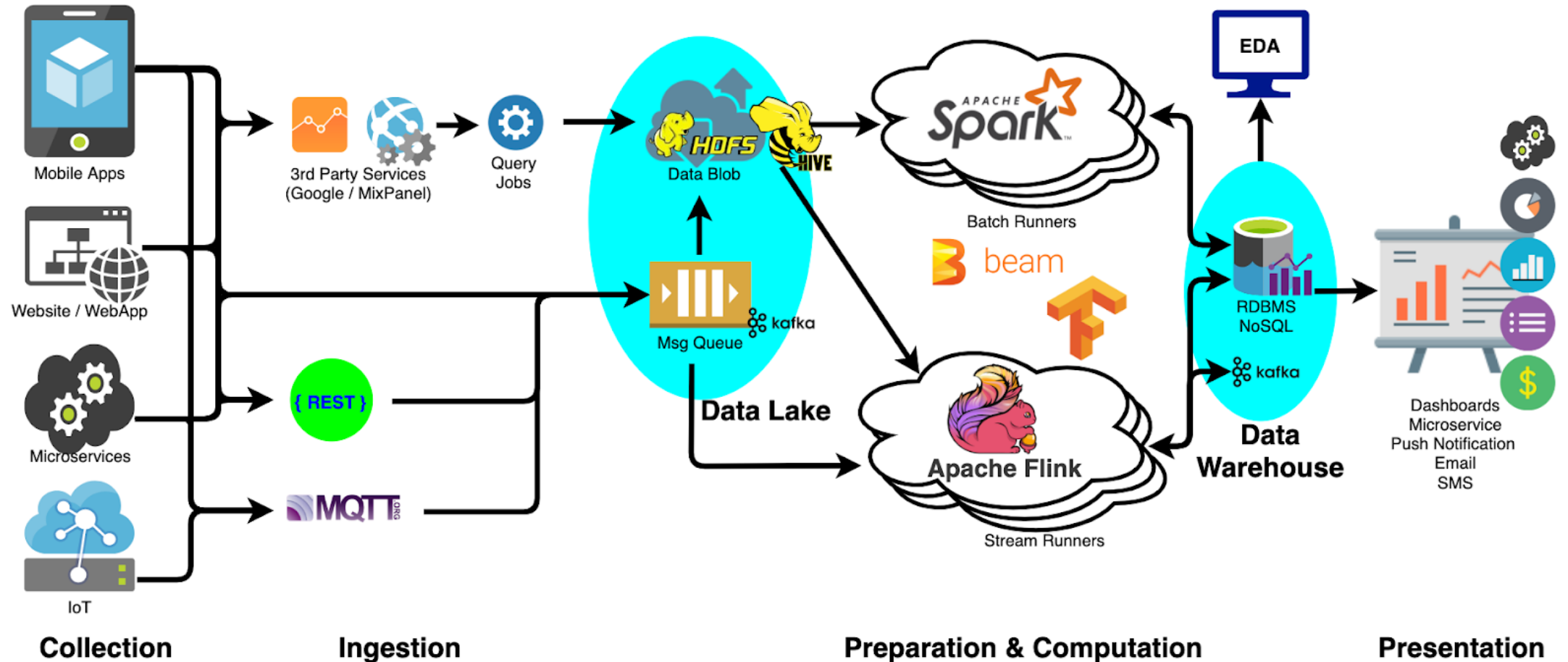
Big data is a collection of data from various sources, often characterized by what's become known as the 3Vs: *volume*, *variety* and *velocity*. Over time, other Vs have been added to descriptions of big data:

VOLUME	VARIETY	VELOCITY	VERACITY	VALUE	VARIABILITY
The amount of data from myriad sources.	The types of data: structured, semi-structured, unstructured.	The speed at which big data is generated.	The degree to which big data can be trusted.	The business value of the data collected.	The ways in which the big data can be used and formatted.
					

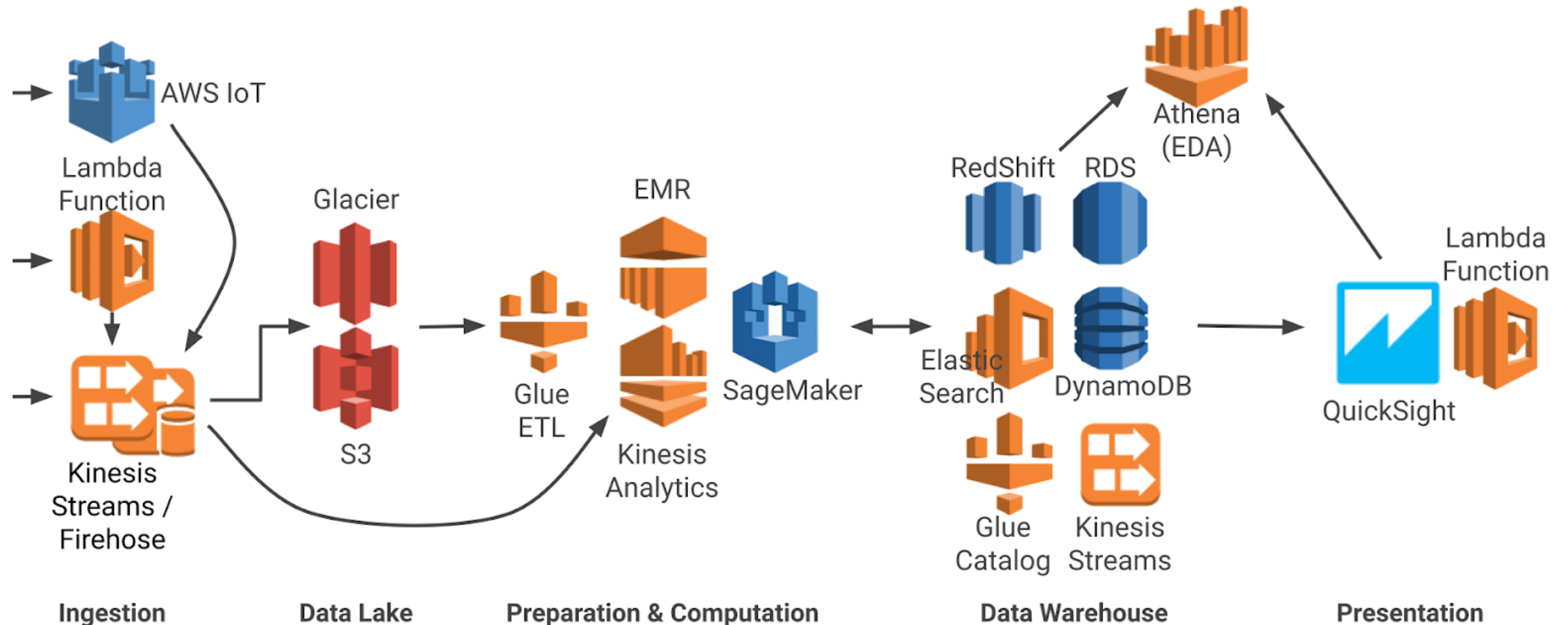
Big Data Architecture: Your choice of the stack on the cloud



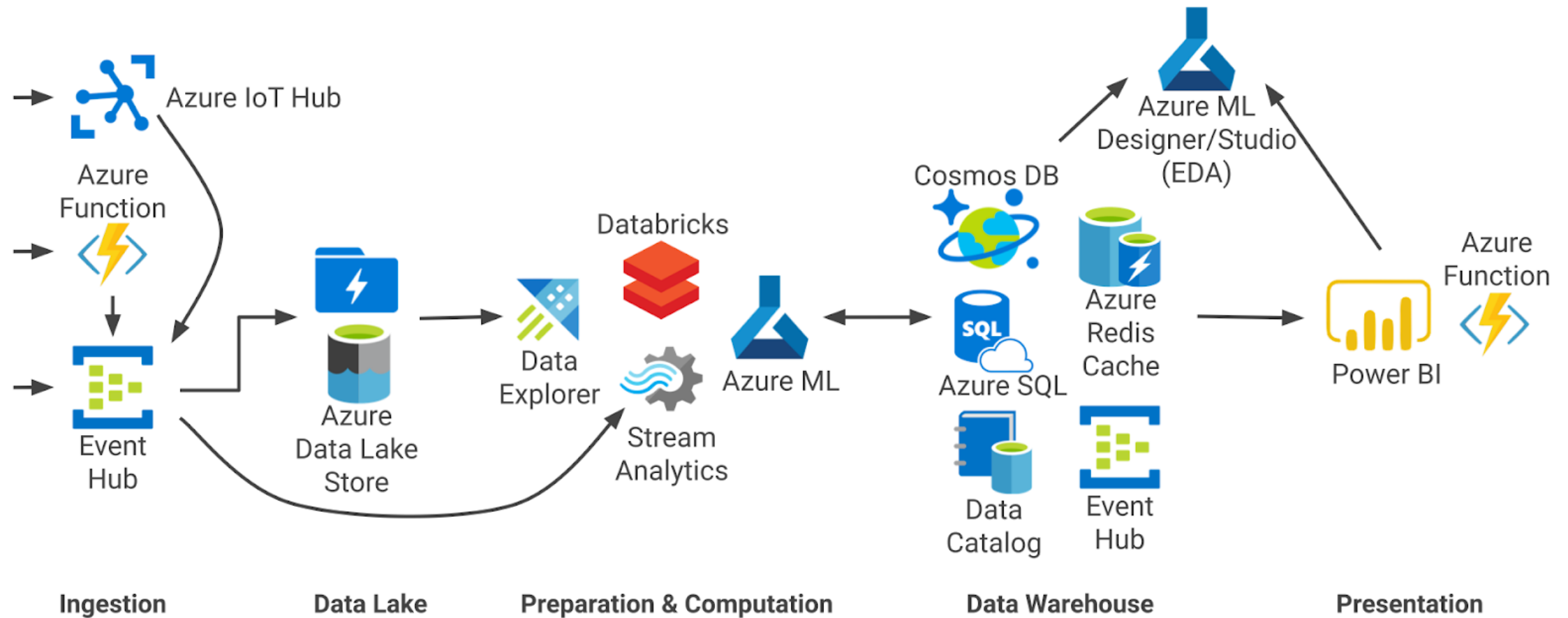
An architecture of the data pipeline using open source technologies



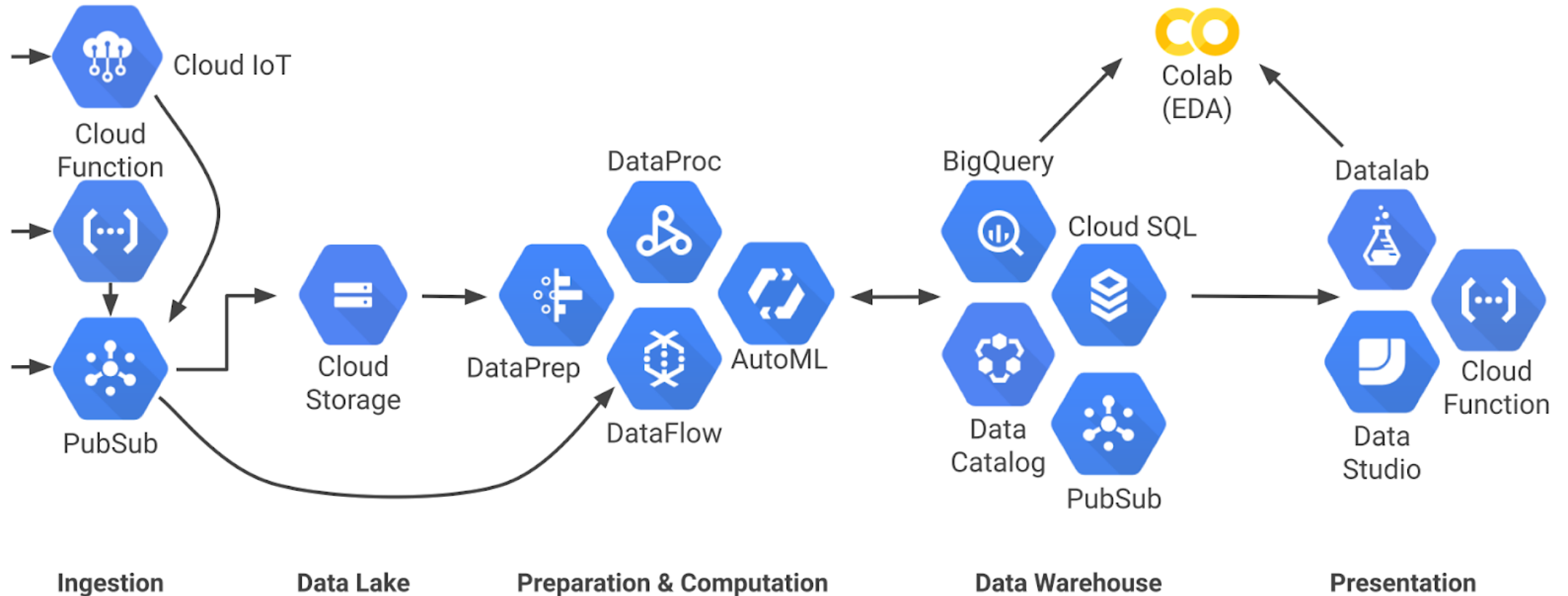
Serverless big data pipeline architecture on Amazon Web Services (AWS)



Serverless big data pipeline architecture on Microsoft Azure



Serverless big data pipeline architecture on Google Cloud Platform (GCP)



large-scale data processing Library

- Dask DataFrame — Flexible parallel computing library for analytics
- PySpark — A unified analytics engine for large-scale data processing based on Spark.
- Koalas — Pandas API on Apache Spark. <https://koalas.readthedocs.io/en/latest/index.html>
- Vaex — A Python library for lazy Out-of-Core dataframes. <https://vaex.readthedocs.io/en/latest/>
- Turicreate — A relatively clandestine machine learning package with its dataframe structure — SFrame, which qualifies.
- Datable — The backbone of H2O's Driverless.ai. A dataframe package with specific emphasis on speed and big data support for a single node.
- H2O — The standard in-memory dataframe is well-rounded. Still, with the recommendations of a cluster four times the size of the dataset, you need deep pockets to use it for exploration and development.
- cuDF (RapidAI) — A GPU dataframe package is an exciting concept. For big data, you must use distributed GPUs with Dask to match your data size, perfect for bottomless pockets.
- Modin — A tool to scale Pandas without changes to the API which uses Dask or Ray in the backend.

Architecture Of Giants: Data Stacks

<https://keen.io/blog/architecture-of-giants-data-stacks-at-facebook-netflix-airbnb-and-pinterest/>

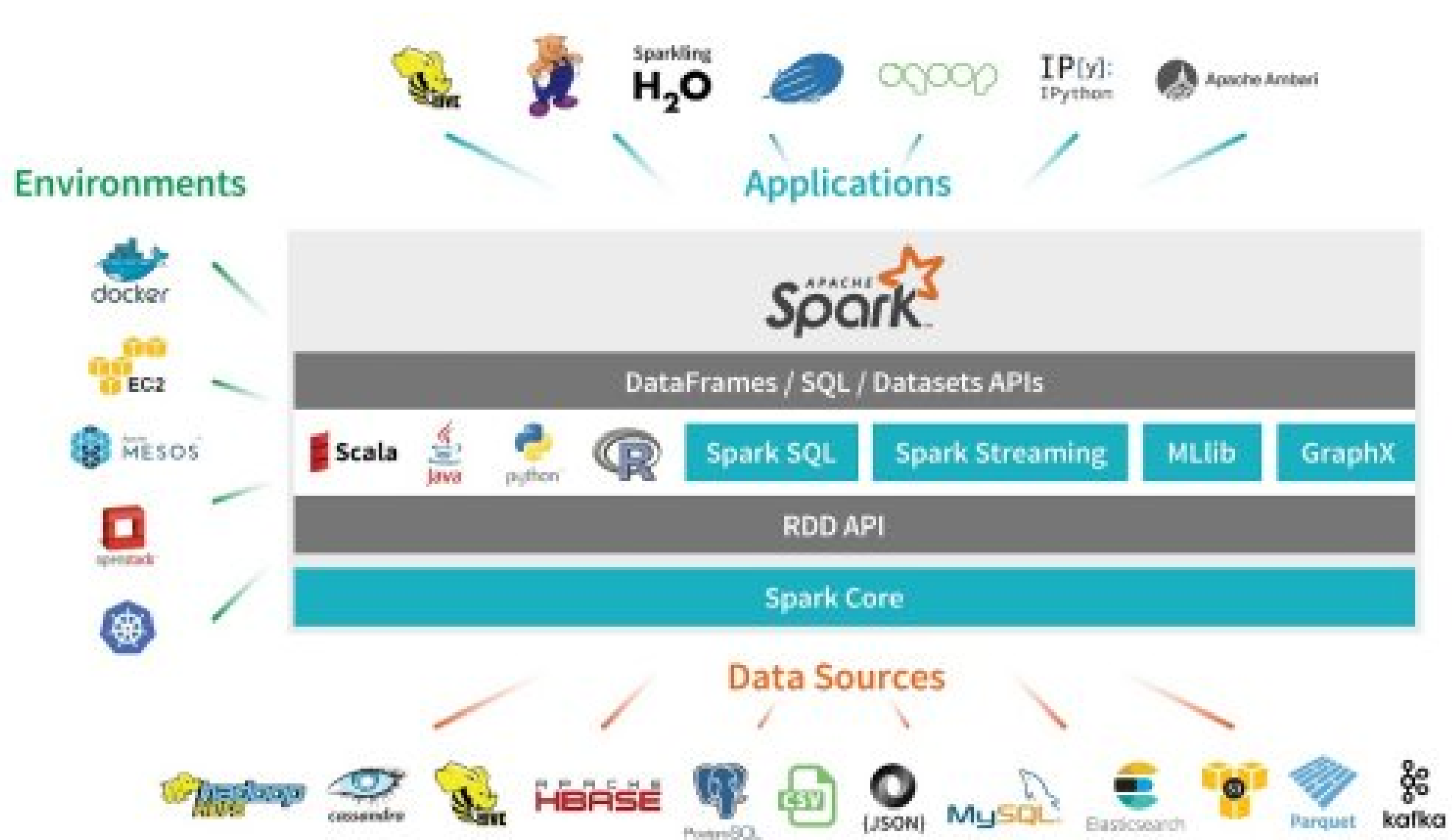
Apache Spark

Apache Spark is an open-source unified analytics engine for large-scale data processing.

Why Apache Spark?

- [Apache Spark](#) is the most [active open source](#) data processing engine built for speed, ease of use, and advanced analytics, with over 1000 contributors from over 250 organizations and a growing community of developers and users.
- Second, as a general purpose compute engine designed for distributed data processing at scale, Spark supports multiple workloads through a unified engine comprised of Spark components as libraries accessible via APIs in popular programming languages, including Scala, Java, Python, and R
- it can be deployed in different environments, read data from various data sources, and interact with myriad applications.

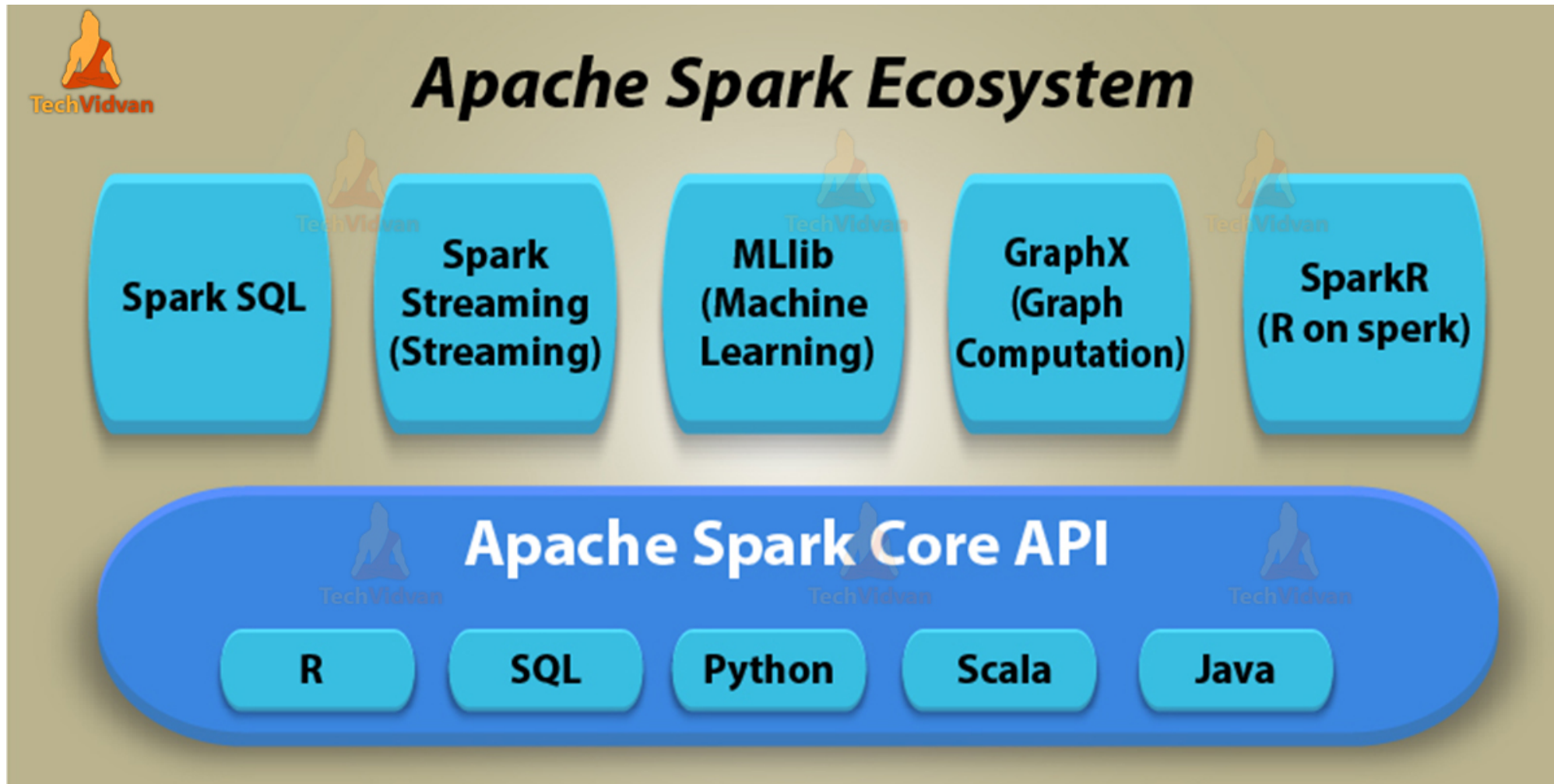
Apache Spark



feature of Spark



Apache Spark Ecosystem Components



PySpark

PySpark is **an interface for Apache Spark in Python**. It not only allows you to write Spark applications using Python APIs, but also provides the PySpark shell for interactively analyzing your data in a distributed environment.

PySpark is **a great language for data scientists to learn because it enables scalable analysis and ML pipelines**. If you're already familiar with Python and Pandas, then much of your knowledge can be applied to Spark.

Environment

There's a number of different options for getting up and running with Spark:

- **Self Hosted:** You can set up a cluster yourself using bare metal machines or virtual machines. Apache Ambari is a useful project for this option, but it's not my recommended approach for getting up and running quickly.
- **Cloud Providers:** Most cloud providers offer Spark clusters: AWS has EMR and GCP has DataProc. I've blogged about DataProc in the past, and you can get to an interactive environment quicker than self-hosting
- **.Vendor Solutions:** Companies including Databricks and Cloudera provide Spark solutions, making it easy to get up and running with Spark.

Practical

Databricks or Google Colab