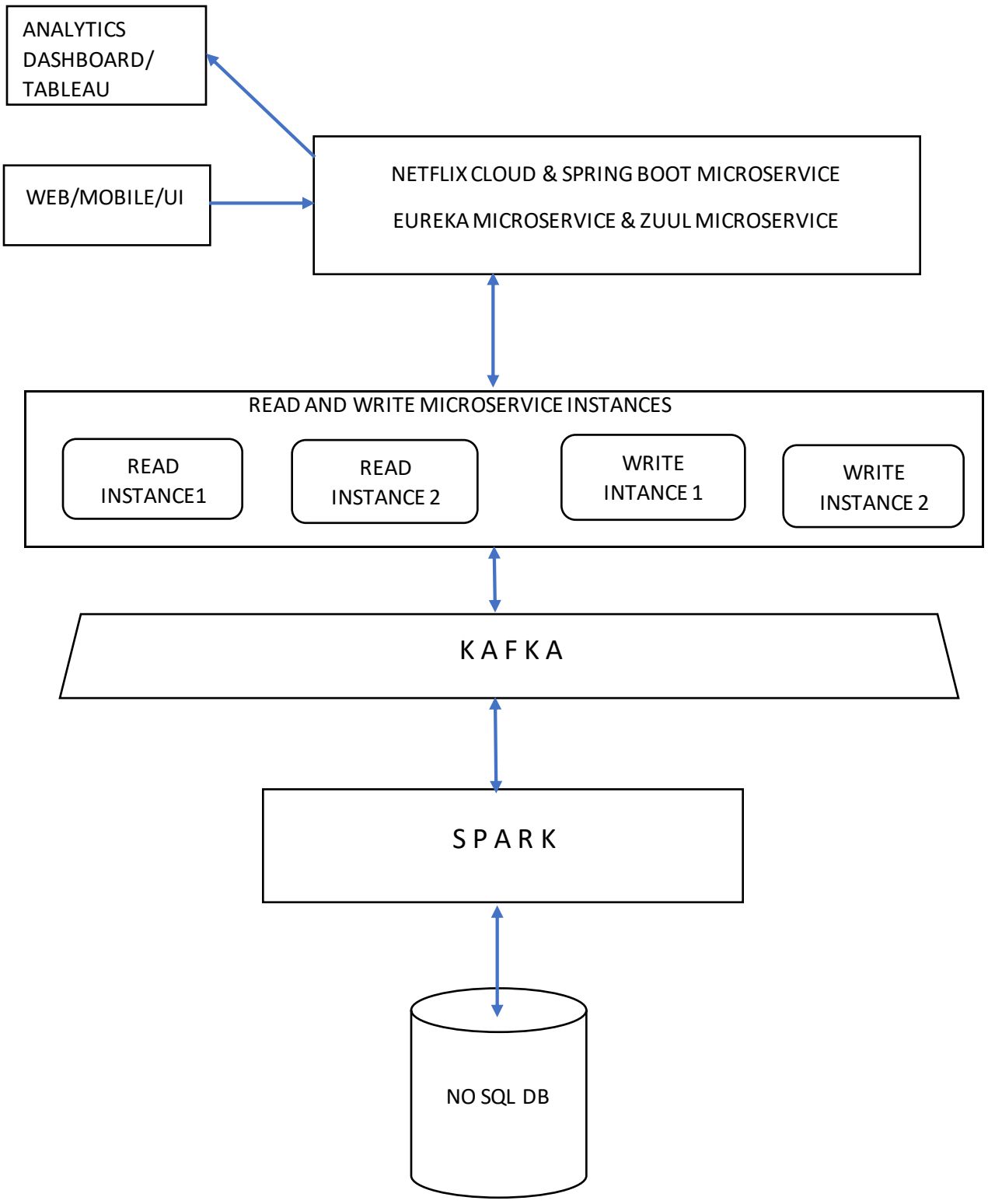
High level design solution for Google Analytic like backend system

We can achieve this goal with numerous components with different technologies/frameworks. Micro services enable us to elastically scale horizontally in response to incoming network traffic into the system. And a distributed stream processing pipeline scales in proportion to the load.



Components:

Web/Mobile User Tracking

Just like Adobe Analytics uses satellite.js to track the user's information with an embedded javascript, similarly, the web pages or mobile sites tracked by Google Analytics embed tracking code that collects data about the user. That data can be stored for monitoring purpose and it's basically javascript which may be used to assign cookie if not set and also sends XHR requests to backend.

Netflix Cloud Architecture with Spring boot microservices

The request can then be handled by Eureka and Zuul microservices. Eureka is for service discovery and Zuul is for proxying, routing and load balancing, they are from Netflix Cloud platform. We can additionally have a load balancer to route requests to the zuul node if an even higher availability needs to be achieved.

Zuul integrated with Netflix component hystrix, for fault tolerance is a circuit breaking mechanism. Zuul is an API gateway that routes traffic based on URI paths providing a layer of security, it auto discovers the services registered in Eureka server.

Eureka allows all clients to register themselves and be used for service discovery.

For scalability and fault tolerance, we use microservices, and can have multiple read and write microservice instances based on the volume of requests in a distributed environment.

Apache Kafka

The write microservices will ingest data into Kafka streams data pipeline through publishing to topics/streams based on the events fired. The write microservice will read it from the stream.

Kafka is scalable with low latency and is run as a cluster on servers/brokers which can be on multiple datacenters across geographies. It uses zookeeper to store metadata about brokers, topics and partitions.

Apache Spark

Data streams can be created from Kafka and then the data is processed and transformed using Apache Spark. It can achieve high performance for both batch and streaming data, has high throughput and fault tolerance.

No SQL Database

The data can be now persisted in a NO SQL database like Cassandra. A layer of in memory database can be added using Spark ignite if required to share the data among jobs before transferring the data into Cassandra. Cassandra is scalable and distributed and can be deployed as a cluster with multiple nodes. Has best write and read performance.

Dashboards For Analytics

The read requests will go from Analytics dashboard through microservices. Apache Spark will do processing of time series data and the results will be sent across to the dashboard for visualization and reporting through microservices. Tableau is a widely used dashboard for analytics and has a variety of pie/donut charts, histograms to analyze and compare, create reports and store historic data.