



WAVESTONE

GRIMALDI STUDIO
LEGALE



TNO



KOMIS

Pilot on **Fair and equal data sharing for cooperative, connected and automated mobility**

Microsoft Azure Components and Data Demonstration

Vagelis Karakolis
ICCS-NTUA

Big Data and B2B platforms: the next big opportunity for Europe
EASME/COSME/2018/004

EASME - European Commission
Executive Agency for Small and
Medium-sized Enterprises



Brussels
17 Sept 2019



Provide the
business case for
“fair and equal”
data sharing for
cooperative,
connected and
automated
mobility

-
The proposed
technical
solution



Items

Shared Server Architecture

Shared Server Azure Components Diagram

Transition from the initial architecture to the final one

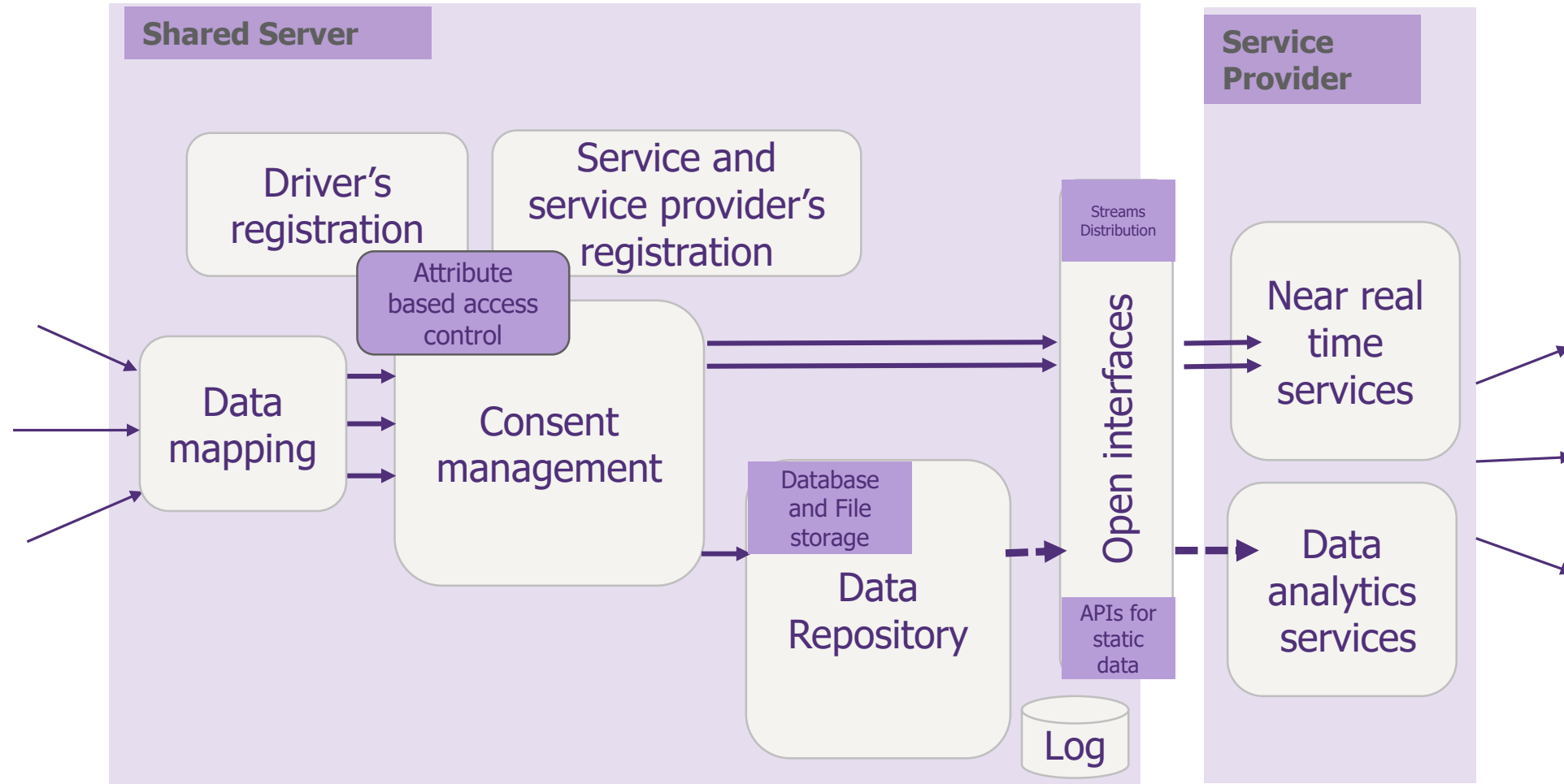
Basic Azure components

Data Demonstration

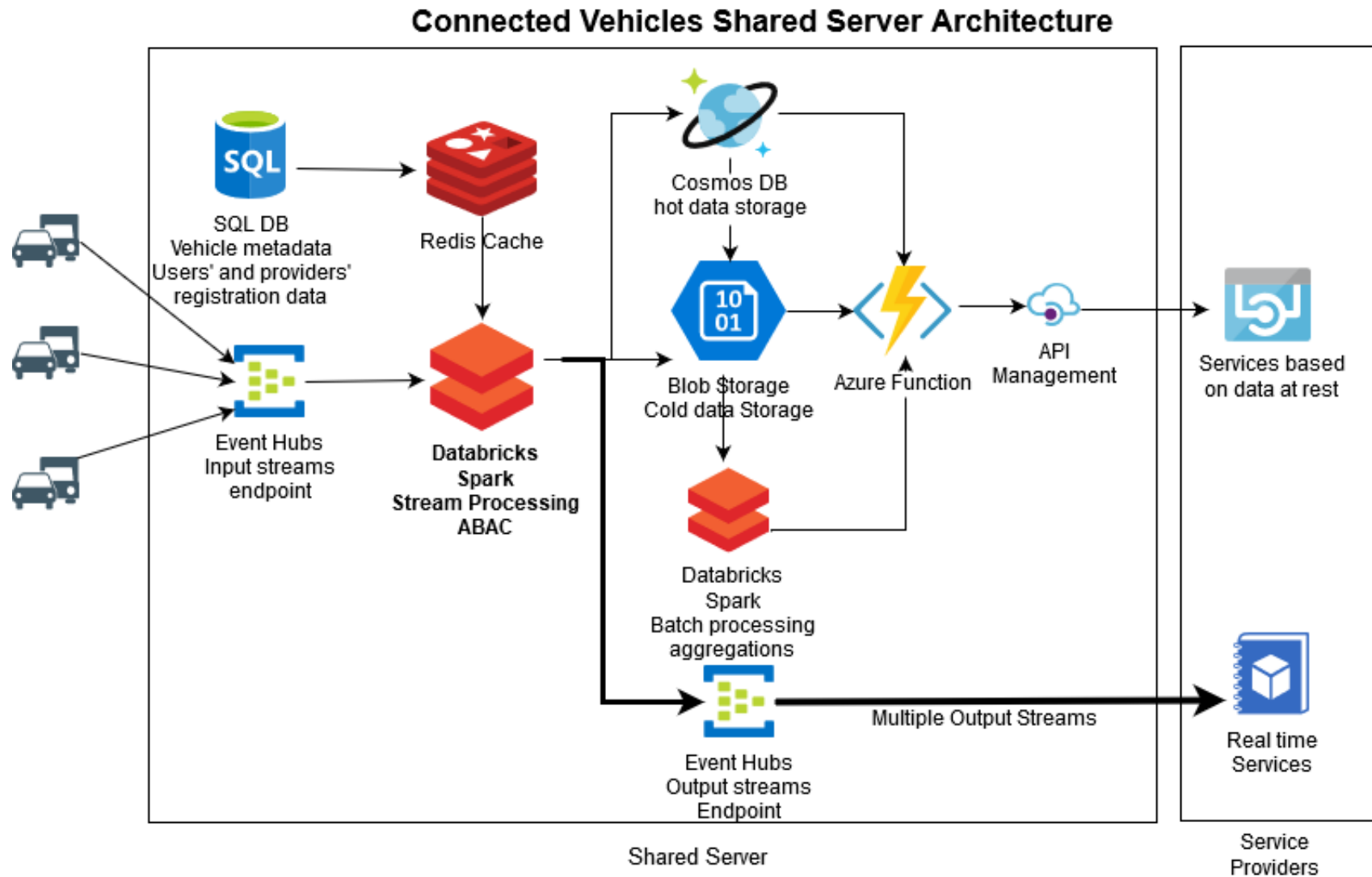
Data Access



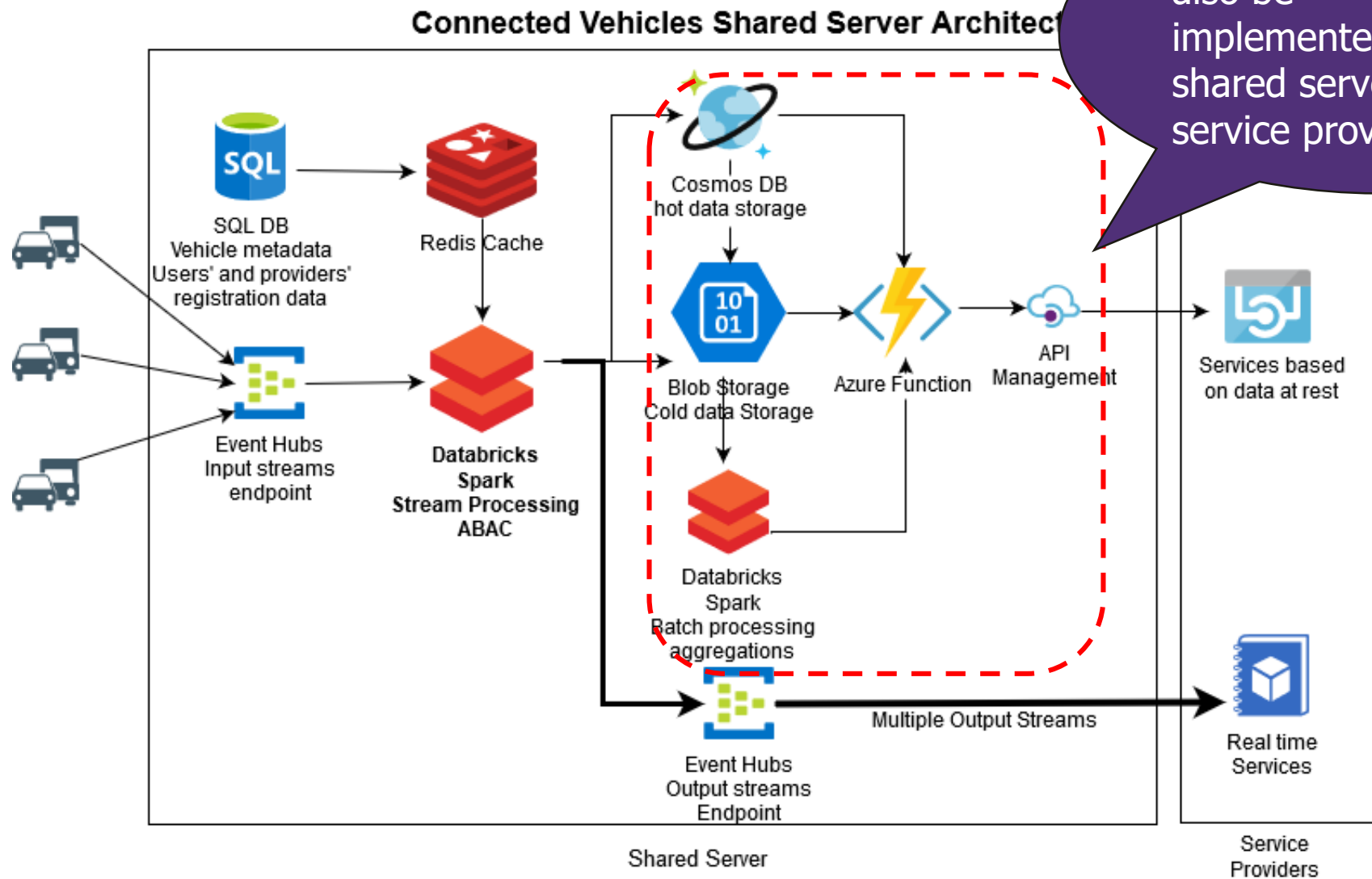
Shared server Architecture



Shared Server Azure Components Diagram



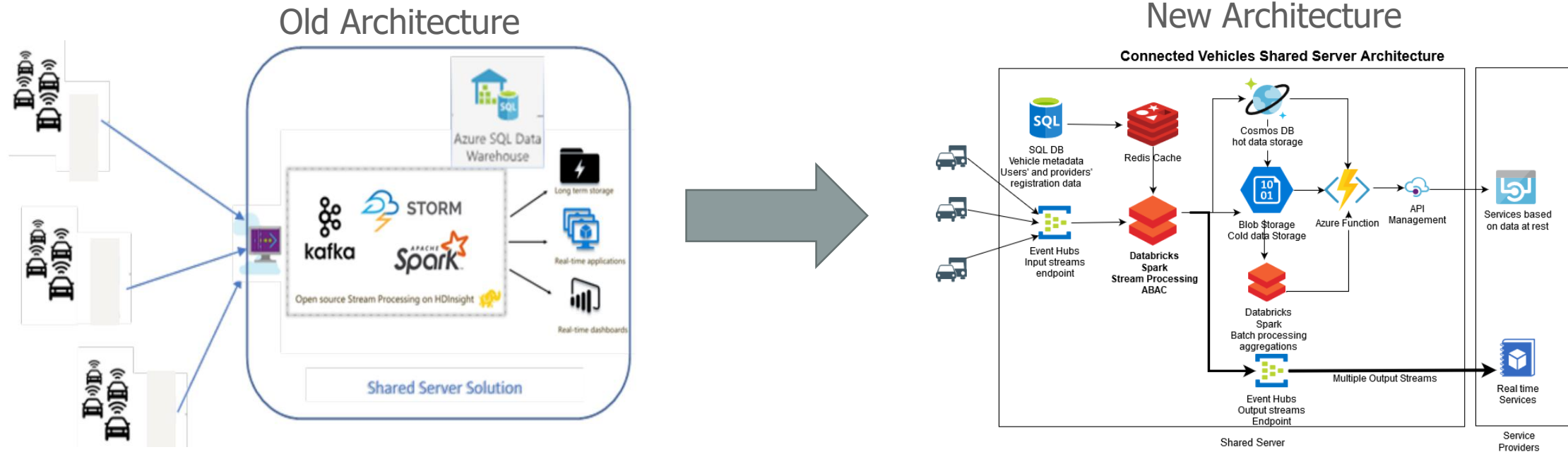
Shared Server Azure Components Diagram



Data Repository can also be implemented outside the shared server from a service provider



From the initial concept to a robust architecture based on Azure Components



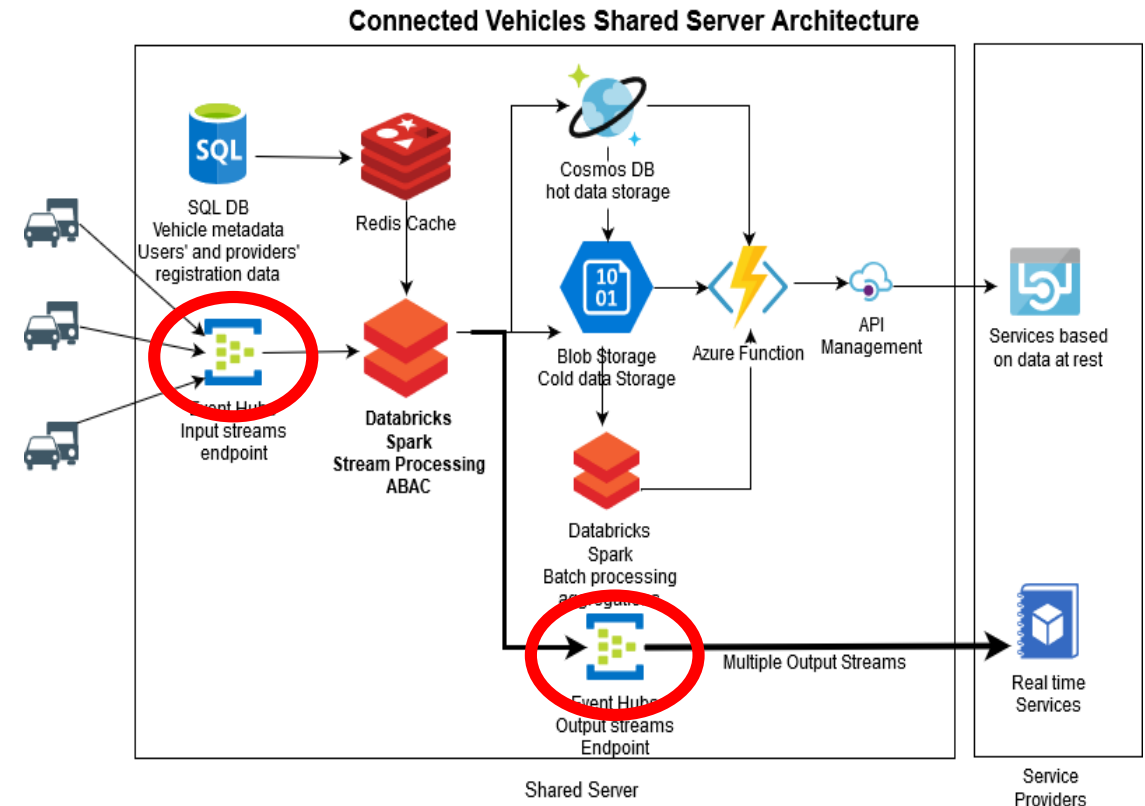
- No need for Kafka
- The role of Apache Storm will be played by Databricks Spark (Spark Streaming)
- Instead of HDInsights Spark we used Databricks Spark, as we did not need any other component of HDInsights because:
 - Databricks is cheaper than HDInsights Spark
 - Databricks is optimized for Spark
 - Databricks provides more and better ways to program Spark and is easier to test
 - Databricks scales more easily than HDInsights Spark
 - Databricks is easier to setup and manage
- Instead of Azure SQL Data Warehouse we used Azure Cosmos DB because the second is:
 - Faster and globally distributed
 - We do not need a relational schema
 - Less probable to fail since it replicates the db to other places



Azure technology is indicative and does not exclude other platforms to be used by the service providers!

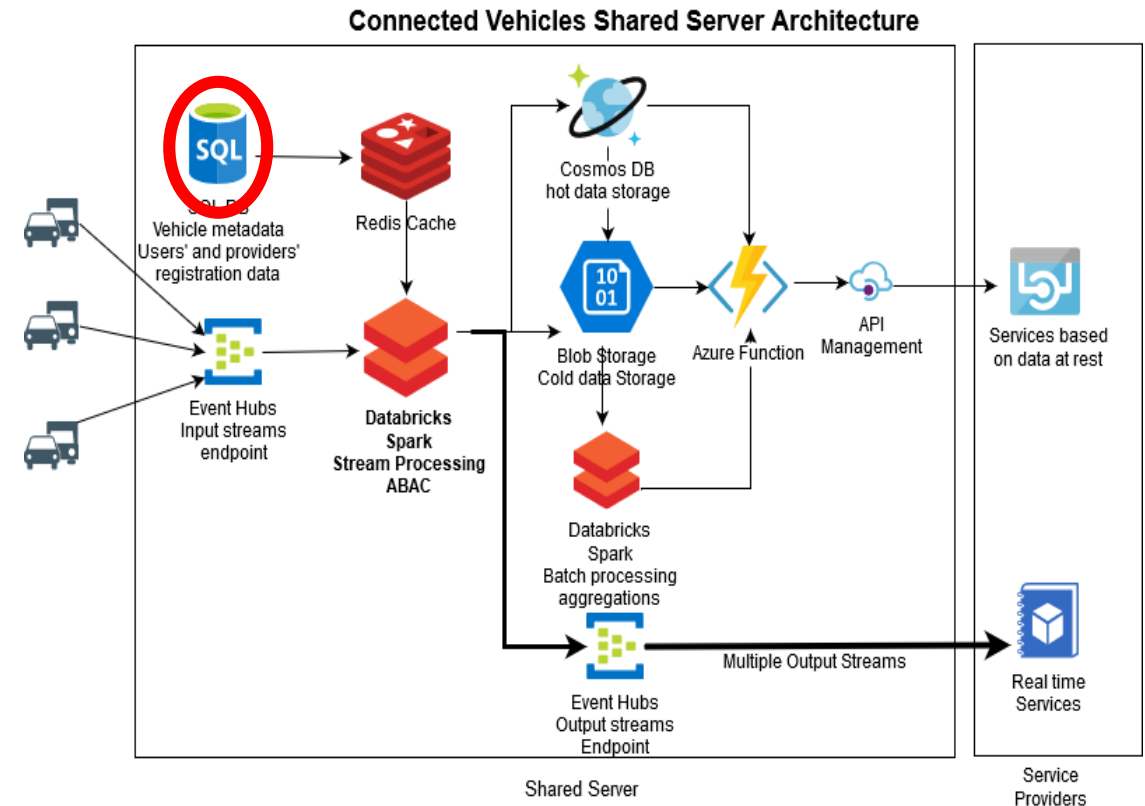
Azure Event Hubs

- Event ingestion service
- Capable of receiving and processing millions of events per second
- The first Event Hubs component will be the endpoint with which the connected vehicles send streams to the shared server
- The second Event Hubs component will be the endpoint with which the services receive data streams from the shared server in the correct form and according to users' consent



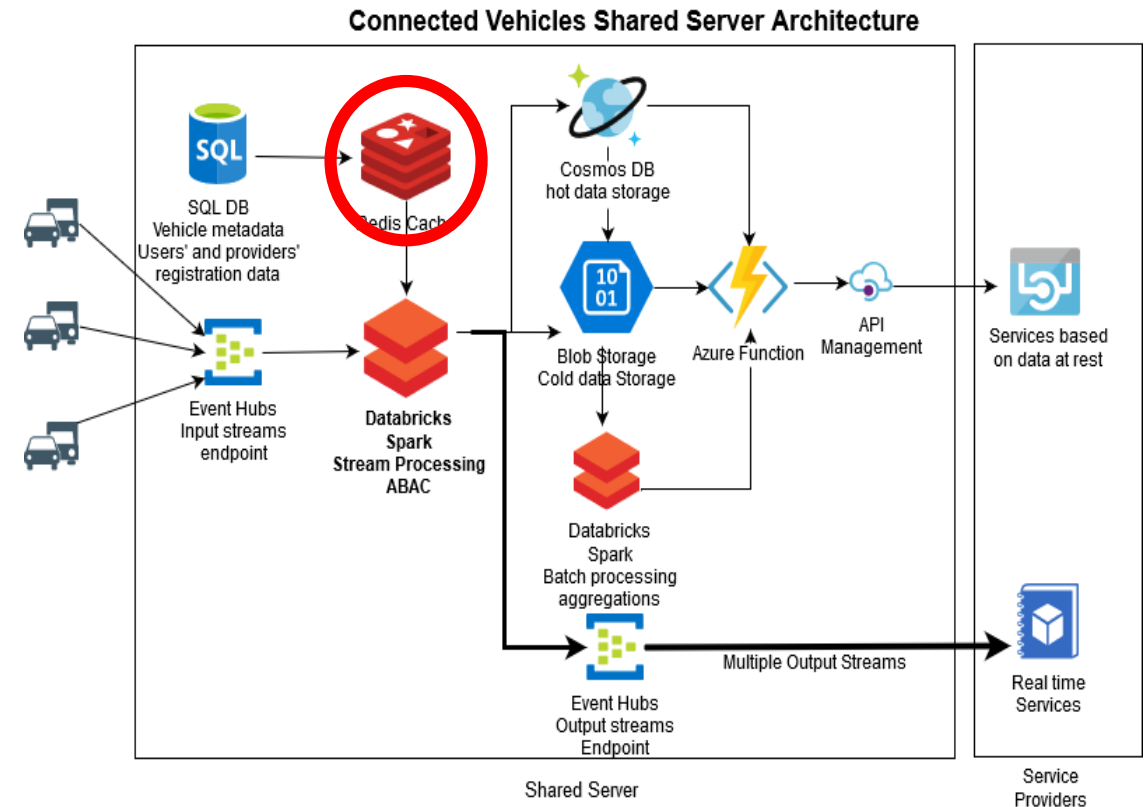
Azure SQL Database

- Cloud database service
- Provides the broadest SQL Server engine compatibility
- In this service the metadata of the users and the service providers will be stored, including users' consent and required data by each service



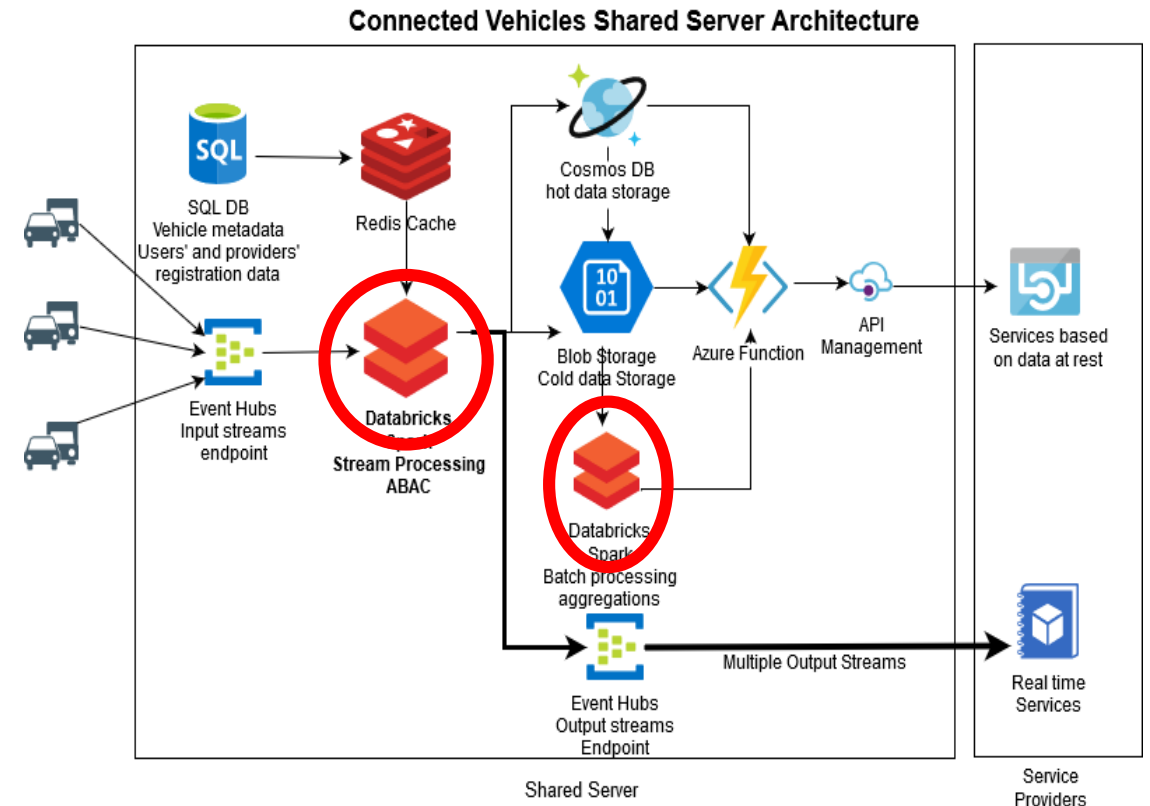
Azure Cache for Redis

- Fully managed, open source, compatible, in-memory data store to power fast, scalable applications
- In our project it will be used to store in memory users' and service providers' metadata in order to be used for Attribute Based Access Control (ABAC)
- It will be updated automatically when there will be database updates



Azure Databricks

- Fast, easy and collaborative Apache Spark-based analytics service
- The first cluster will receive the streams of telemetry data and, after processing, it will provide multiple stream outputs that take into consideration the vehicle owners' consent and will be consumed by different real time services
- It will also provide outputs for storing the data to Cosmos DB and Azure Blob Storage
- The second cluster will perform batch processing to the Azure Blob Storage, which is file storage built on HDFS, in order to extract aggregate values from data
- The second cluster will be running periodically



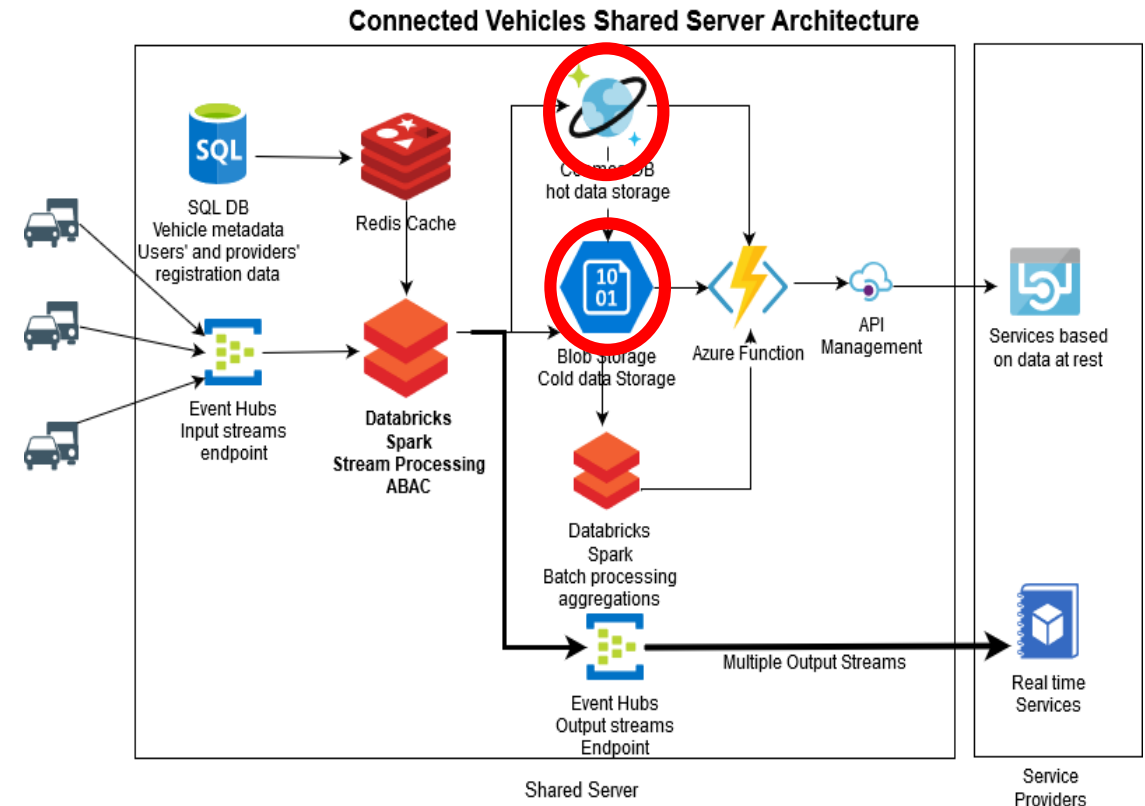
Azure Cosmos DB and Azure Blob Storage

Azure Cosmos DB

- **Globally distributed, fully managed, multi-model database service for any scale**
- **It will be used for storing the fresh telemetry data, in order to be accessed by other services**

Azure Blob Storage

- **Massively scalable object storage for unstructured data**
- **It will be used for storing all the data in order to be used for analysis**



Azure Function and Azure API Management

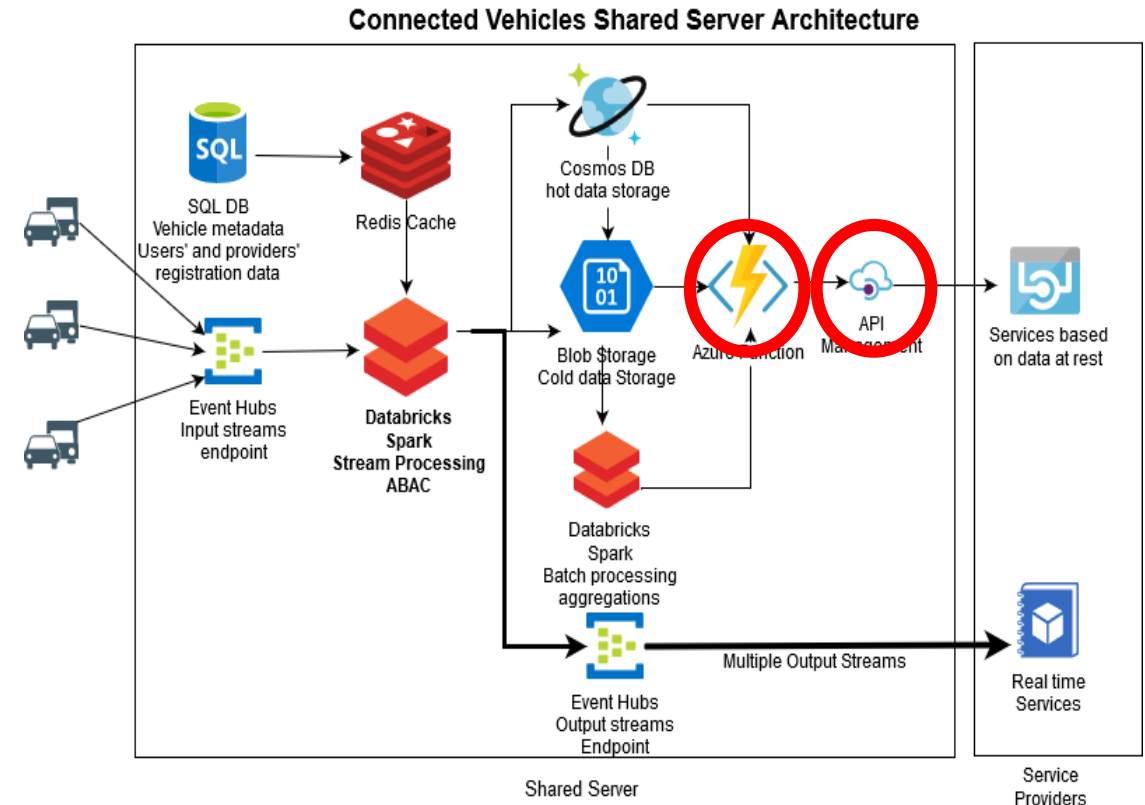
Azure Function

- Serverless compute service that enables you to run code on demand without having to explicitly provision or manage infrastructure
- It will be used to implement serverless web APIs in order to control the service providers' access to the data

Azure API Management

It will be used to:

- Create policies for APIs usage
- Monetize the APIs usage
- Monitor the APIs usage





**Data
Demonstration**





Questions Answers