CALCULATE THE REMAINING LIFE OF HOUSEHOLD APPLIANCES

Faculty Of Science And Engineering

SCALABLE COMPUTING (WMCS16003)

Group 25

Swastik S Nayak

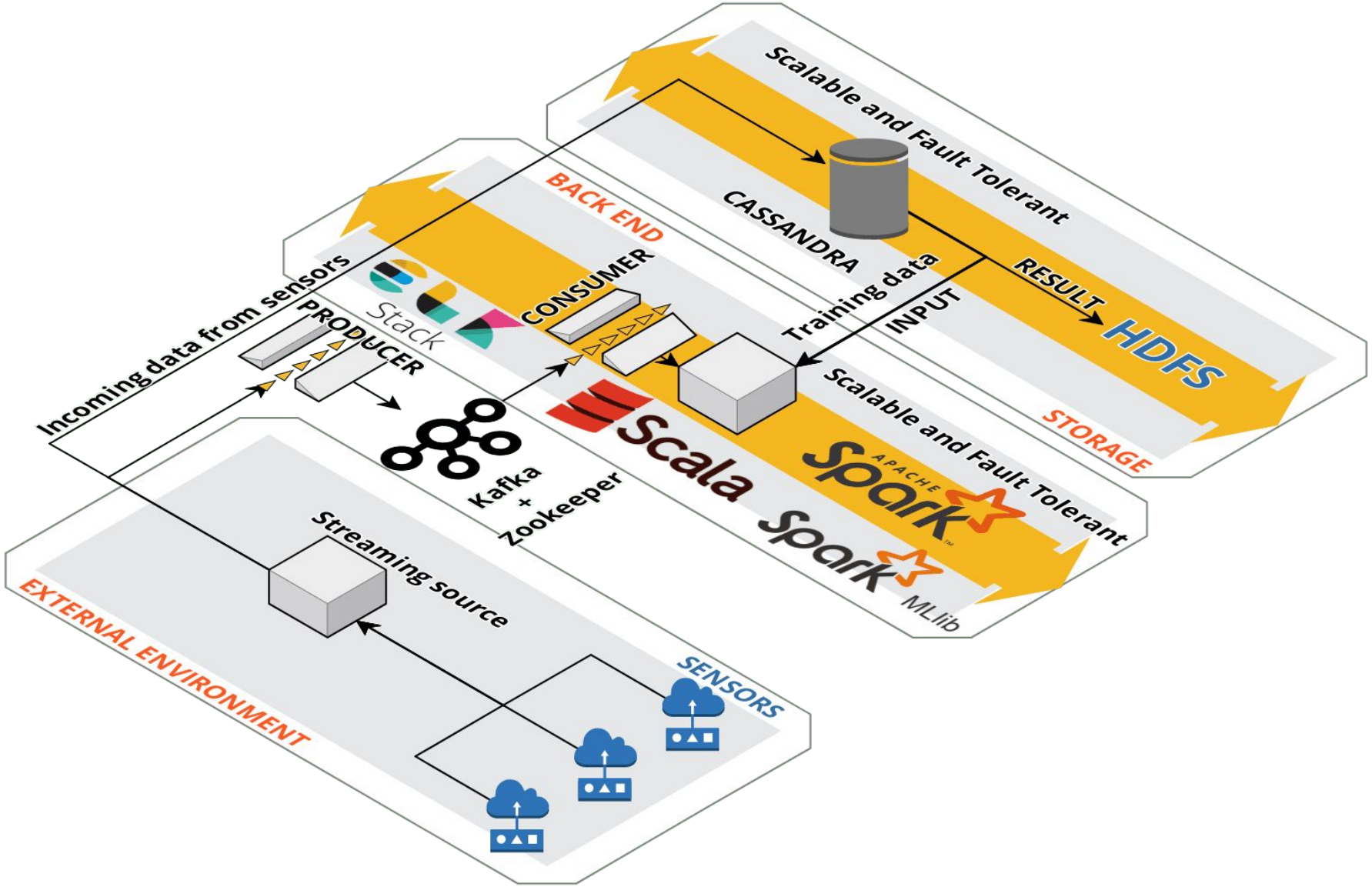
Siddharth B

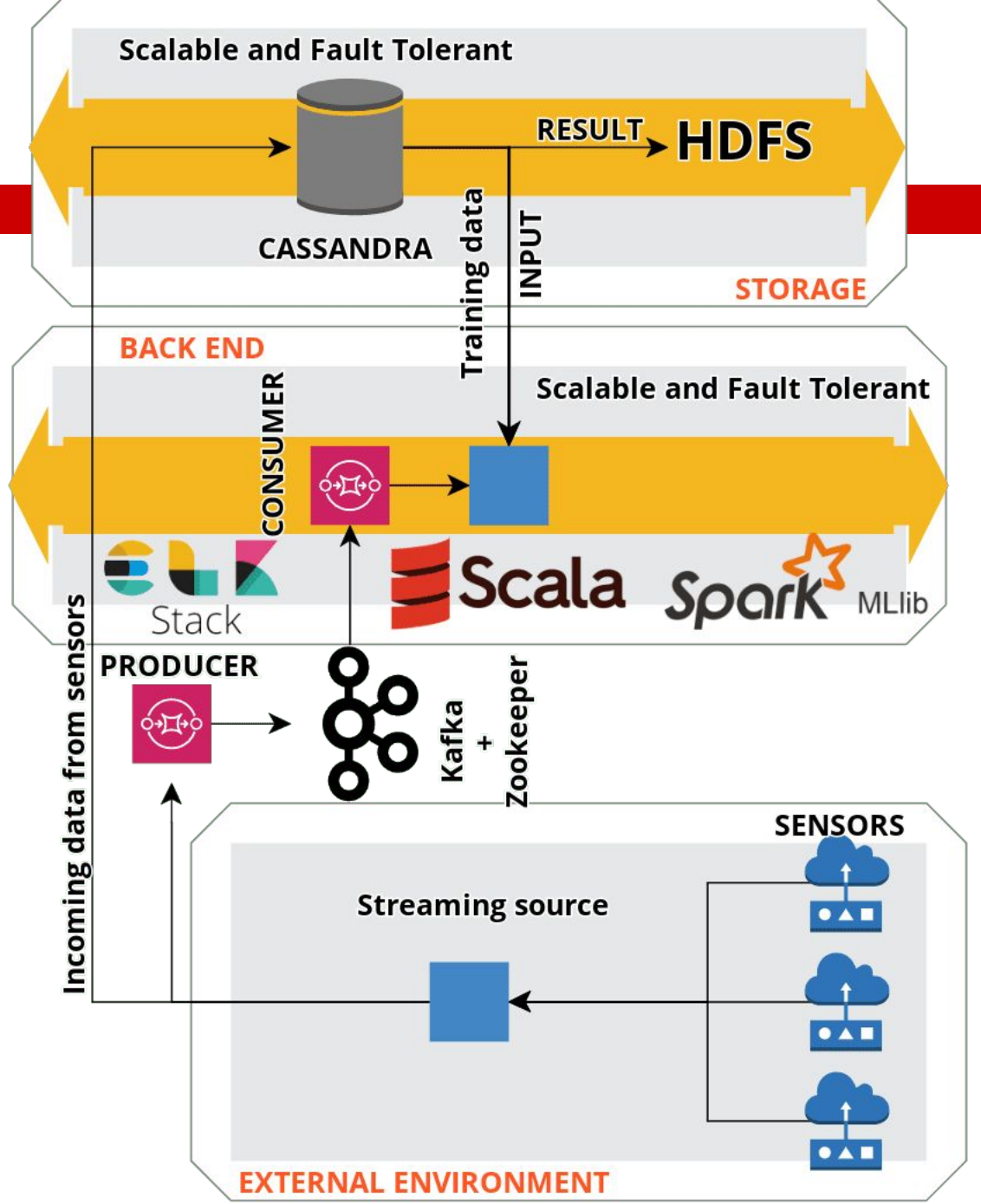
Anil P.Mathew

Project Proposal

- An application that predicts the remaining lifespan of household appliances based on their electrical usage and the age of the appliance.
- A knowledge base should be constructed (training data) that should be utilized to train the model and predict the novel data.
- Sensor data from multiple appliances should be streamed to the application to predict the relevant appliances lifespan.

Architecture




Scalable and Fault Tolerant
RESULT
HDFS
CASSANDRA
Training data
INPUT
STORAGE
BACK END
CONSUMER
Scalable and Fault Tolerant
Stack
Scala
Spark
MLlib
PRODUCER
Kafka
+
Zookeeper
Incoming data from sensors
SENSORS
Streaming source
EXTERNAL ENVIRONMENT

Requirements

FUNCTIONAL

- A batch processing pipeline to train the model.
- Handle the incoming streaming data from the sensors from multiple appliances.
- Algorithm to predict the lifespan of the appliances.

NON-FUNCTIONAL

- Scalability, the cluster should scale based on incoming data.
- Fault-tolerant, the application should recover from unforeseen errors.
- Performance, the application should predict the outcome in

Technologies

TECHNOLOGY	KEY FEATURES	USAGE
Scala	Scalable or Multi-paradigm (FP+OOP), concise, strongly typed, immutable-first language, non existent boiler plate codes, interoperability with java, easy to maintain, leverage the advantages of JVM.	The backend code will consist mainly of scala to enable a scalable, concise, clean and maintainable code. The language will allow for a faster turnaround time to reduce the coding efforts. Will contain the business logic or the Engine which drives the web application.
HDFS	Fault tolerance, high availability, high reliability, replication, scalability and, distributed storage.	Storage used to support the distributed computation in the application.
Spark	Speed, Supports multiple languages, In-memory computation, Advanced analytics, SparkSql and SparkML.	Spark will be used to execute the application in distributed fashion. It will utilize HDFS and Yarn to distribute the work across the cluster.
Kafka	High Throughput, lower latency, fault tolerant and durability.	Deals with streaming of data across the system. Provides a direct stream to the backend, generic kafka stream reader.
Cassandra	Handles massive amount of unstructured data, high availability and fault tolerant, elastic scalability, high performance.	Secondary database to create a fault tolerant environment. Structure: Key1, Key2, Value
Docker	Compatibility and maintainability, simplicity and faster configuration, rapid development, deployment and testing.	Used to simulate the environment and create docker images for continuous deployment, testing and releases.

TECHNOLOGY	KEY FEATURES	USAGE
ELK **(optional)**	Elastic search to support RESTful search and analytics. Logstash to support server-side data processing pipeline that ingests, transforms and loads data. Kibana lets you visualize your Elasticsearch data and navigate the Elastic Stack.	Used for tracking the status of input files and other metrics.
SBT	Build tool.	To handle continuous integration, profiling, and dependencies across the project.

rijksuniversiteit
groningen

Thank You