Project Athena

Done by Jakob Neuhauser, Ivonne Simic and Paul Pavlis

# Table of contents

[1. Table of contents 1](#_Toc42895580)

[2. Executive Summary 2](#_Toc42895581)

[3. System Architecture 3](#_Toc42895582)

[Diagram 3](#_Toc42895583)

[Procedure & Technologies 3](#_Toc42895584)

[Summary of the overall procedure 3](#_Toc42895585)

[Apache Nifi 4](#_Toc42895586)

[Elasticsearch / Kibana 4](#_Toc42895587)

[Hadoop HDFS & Apache Spark 4](#_Toc42895588)

[Apache Kafka 4](#_Toc42895589)

[Greenhouse Data 5](#_Toc42895590)

[Data Picture 5](#_Toc42895591)

[Description 5](#_Toc42895592)

[4. Steps to reproduce (from readme.txt) 6](#_Toc42895593)

# Executive Summary

# System Architecture

## Ein Bild, das Text, Karte enthält. Automatisch generierte BeschreibungDiagram

## Procedure & Technologies

### Summary of the overall procedure

To start the whole process, we need to receive the sensors data via messages, so that we can transform and use them for the following steps. This is being achieved with Apache Nifi. The Program listens on a specific port and each sensor data entry that is sent there, will be converted into a usable file (flowfile in Nifi). These flowfiles then undergo a check and are transformed to the right json format for later use.

We use these sensor data sets for three different workflows:

* (Near) Real-Time Analysis and Dashboard
  + The data is transported (with Apache Nifi) to Elasticsearch, where it will be stored at an index (index can basically be associated with a table in SQL). From there the data will be accessed with Kibana and mapped onto different Graphs. These Graphs show the flow of the current (and near past) sensor data and the different values for e.g. the temperature or soil moisture.
* Long-term storage and queries on the dataset
  + First, the data from the different Nifi flowfiles are merged into fewer larger files, so that the following technology is more efficient in handling large amounts of data. The data is then stored in HDFS (Hadoop distributed file system), which basically just stores the data for later use. Then a Query Engine (Apache Spark) is used to make queries on the whole dataset to get e.g. the average temperature of the year 2019. The data from the analysis can be used to correct different aspects of the greenhouse.
* Check for critical values and changing the hardware accordingly
  + In this pipeline, the values of the sensor data are checked (in Nifi) against predefined maximal and minimal values. If the value is above or below the threshold, the file will be sent to a message broker (Apache Kafka). From there the critical values can be extracted and sent to the hardware to change these values on e.g. the flowerpots. (The Hardware aspect is out of scope for this project. The data will only be provided, not used)

### Apache Nifi

### Elasticsearch / Kibana

### Hadoop HDFS & Apache Spark

### Apache Kafka

## Greenhouse Data

### Data Picture

### Description

This part explains the data that is being used for the whole project. You can see a snippet of the data in the previous paragraph.

# Steps to reproduce (from readme.txt)