

Trabajo módulo IV

Procesamiento de datos de calidad del
aire de Madrid.



Análisis del caso de uso y arquitectura planteada



Descripción de caso de uso.

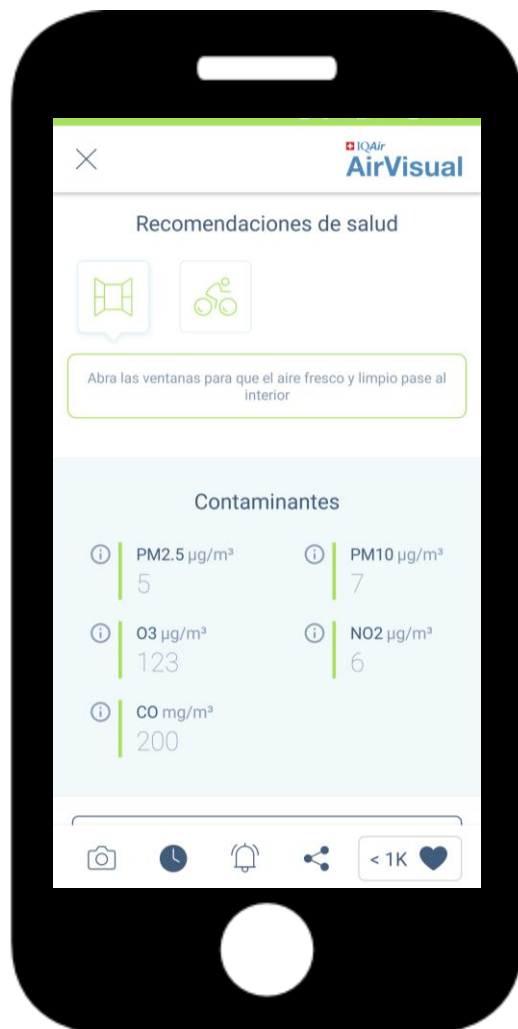
MEDICIÓN DE CALIDAD DEL AIRE DE MADRID

- Analizar los datos proporcionados por ayuntamiento de Madrid.
- Disponibilizar los datos de manera óptima para:
 - ▷ Consulta real-time de la calidad del aire por parte de usuario final.
 - ▷ Consulta analítica por parte de los analistas de la Comunidad de Madrid.

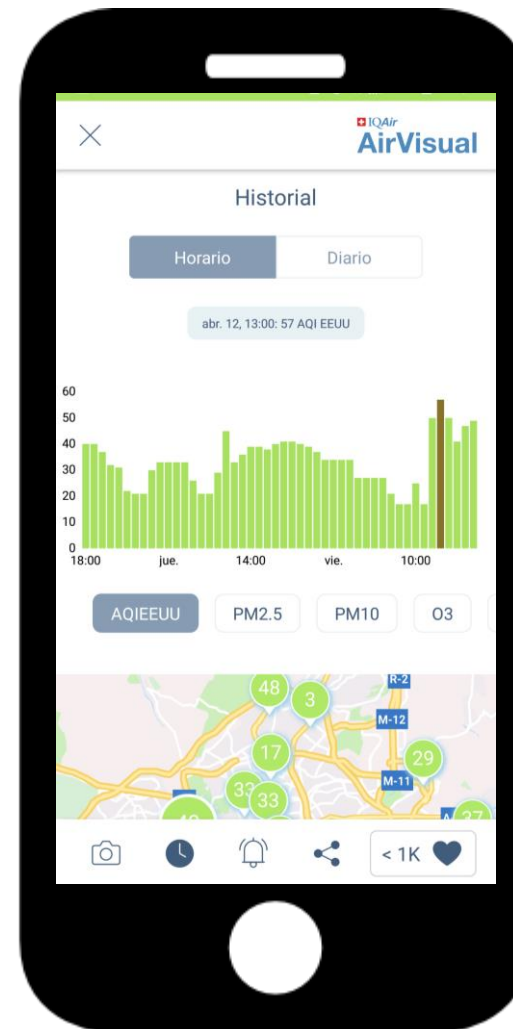




Descripción de caso de uso.



Consulta del estado del aire en el momento.



Ver principales indicadores abiertos por horario y ubicación geográfica.



Datos disponibles



Actualización cada
1 hora

Datos disponibles:

■ Geo Espaciales:

- ▷ Provincia
- ▷ Municipio
- ▷ Estación

■ Temporales:

- ▷ Año
- ▷ Mes
- ▷ Día
- ▷ Hora

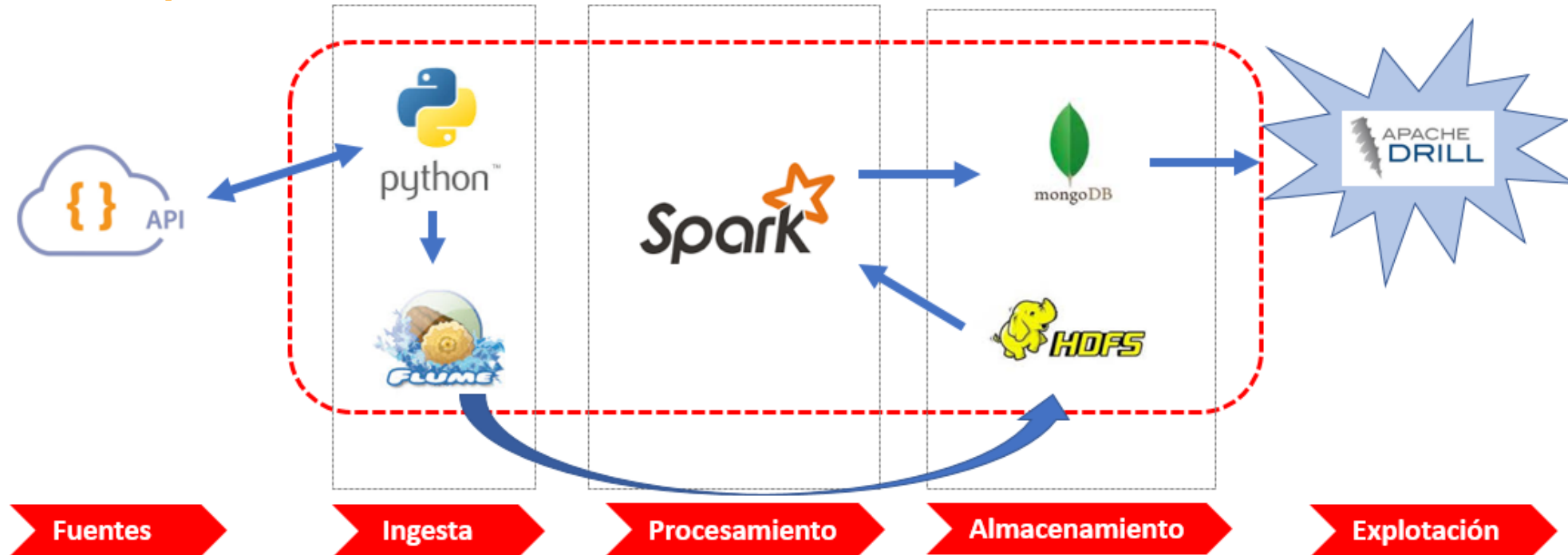
■ Magnitudes de calidad:

- ▷ UV: radiación ultravioleta.
- ▷ DV: dirección del viento.
- ▷ VV: velocidad del viento.
- ▷ RS: radiación solar.
- ▷ HR: humedad relativa.
- ▷ PB: presión barométrica.
- ▷ T: temperatura.
- ▷ P: precipitación.



Arquitectura y flujo de datos

Arquitectura planteada:



A woman in a white shirt and black pants stands on the left, presenting to a group of people seated at long wooden tables. The room features a large, vibrant mural of a forest on the right wall. Several laptops are open on the tables, and the audience is focused on the presentation. The scene is lit with modern, recessed ceiling lights.

Detalles del pipeline



Ingesta

```
# API request:
url = "http://www.mambiente.madrid.es/opendata/horario.csv"

while True:
    # Hacemos la llamada a la API y elegimos el separador del csv
    s=requests.get(url).content
    c=pd.read_csv(io.StringIO(s.decode('utf-8')), sep=';')

    # Creamos las columnas concatenando valor con indicador de validez:
    c['H01'] = c.V01+c.H01.map(str)
    c['H02'] = c.V02+c.H02.map(str)
    c['H03'] = c.V03+c.H03.map(str)
    c['H04'] = c.V04+c.H04.map(str)
    c['H05'] = c.V05+c.H05.map(str)
    c['H06'] = c.V06+c.H06.map(str)
    c['H07'] = c.V07+c.H07.map(str)

    # Borramos las columnas que no nos sirven:
    drop_columns = {'V01', 'V02', 'V03', 'V04', 'V05', 'V06' \
, 'V07', 'V08', 'V09', 'V10', 'V11', 'V12', 'V13', 'V14' \
, 'V15', 'V16', 'V17', 'V18', 'V19', 'V20', 'V21', 'V22', \
'V23', 'V24'}

    c.drop(columns=drop_columns, axis=1)

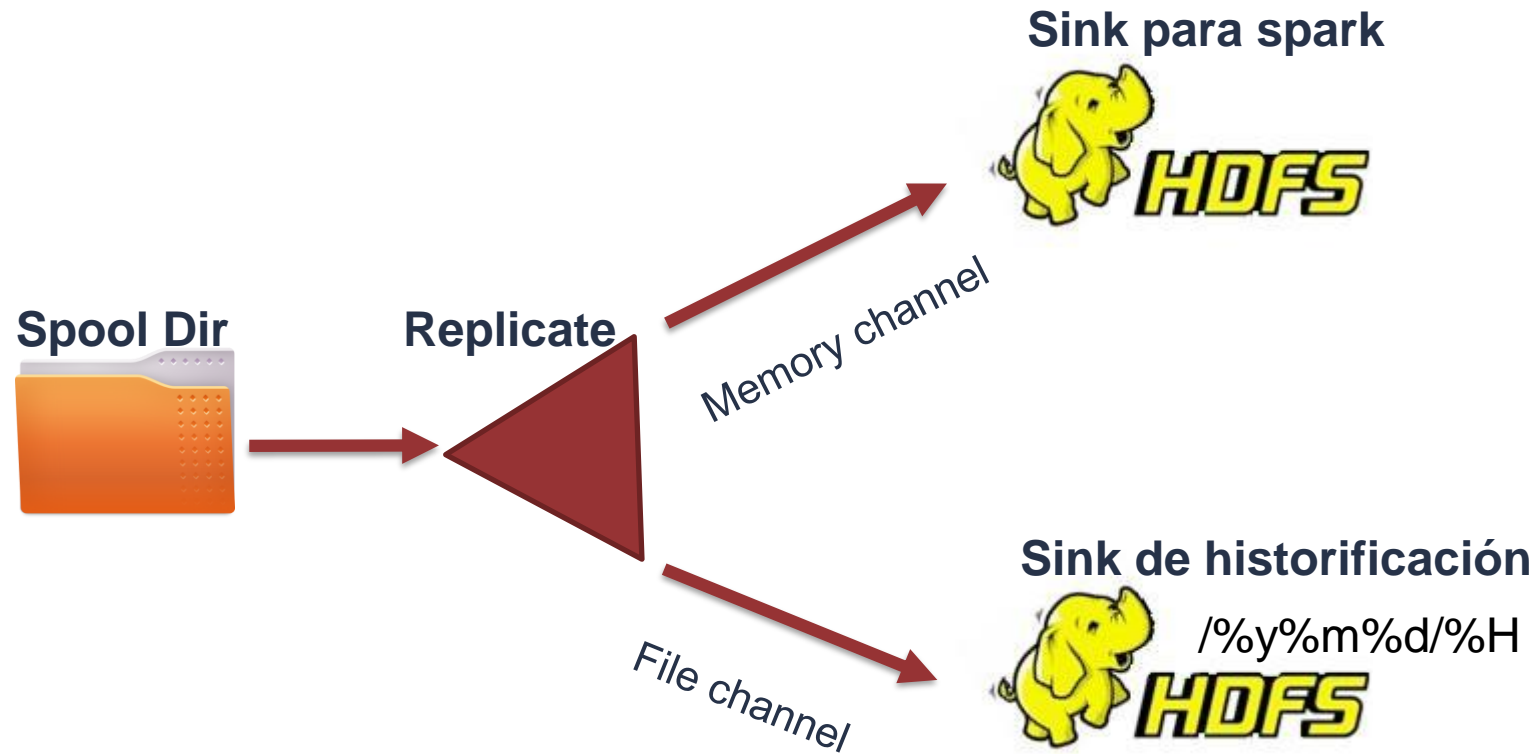
    # Guardamos el fichero:
    eventTime = time.strftime("%Y%m%d%H%M")
    file_name = "airQuality_"+eventTime+'.csv'
    c.to_csv(r'./data/'+file_name, sep='\t',encoding='utf-8', index=False)

    # Se ejecuta el while cada hora
    time.sleep(3600)
```





Ingesta





Procesamiento

```
// Leer la copia del ultimo csv con la medición, las dos fuentes de códigos-estaciones
// y códigos-magnitudes.
val aire = spark
    .read
    .option("header", "true").csv("/user/abalserio/Grupal4_Flume/datosSpark/")
val codigo_estaciones = spark
    .read
    .option("header", "true").csv("./src/main/scala/m4/resources/codigo_estaciones.csv")
val codigo_magnitud = spark
    .read
    .option("header", "true").csv("./src/main/scala/m4/resources/codigo_magnitud.csv")
```

```
// Con Spark SQL se prepara el csv añadiendo la fecha en formato fecha y limpiando los valores de las medidas
aire.createOrReplaceTempView("air")
val air2 = spark.sql("""select provincia, MUNICIPIO, ESTACION, MAGNITUD, PUNTO_MUESTREO, ANO, MES, DIA,
    from_unixtime(unix_timestamp(
        concat(cast(ANO as STRING),
        RIGHT(concat('0',cast(MES as STRING)),2),
        RIGHT(concat('0',cast(DIA as STRING)),2)),
        "yyyyMMdd")) AS _id,
    case when H01 like ('V%') then substring(H01, 2, length(H01)) else null end H01_ ,
    case when H02 like ('V%') then substring(H02, 2, length(H02)) else null end H02_ ,
    case when H03 like ('V%') then substring(H03, 2, length(H03)) else null end H03_ ,
    case when H04 like ('V%') then substring(H04, 2, length(H04)) else null end H04_ ,
    case when H05 like ('V%') then substring(H05, 2, length(H05)) else null end H05_ ,
    case when H06 like ('V%') then substring(H06, 2, length(H06)) else null end H06_ ,
    case when H07 like ('V%') then substring(H07, 2, length(H07)) else null end H07_ ,
```





Procesamiento

```
//Se procede a generar la estructura de lo que será el JSON que luego se guardará en MongoDB
val df1 = aire_tuning2.select(col("Nombre_Estacion"), col("_id").as("Fecha"),
    struct(
        col("PUNTO_MUESTREO"),
        col("MAGNITUD_NOMBRE"),
        struct(
            col("H01_"),
            //<...>
            col("H24_")
        ).as("values")
    ).as("info")
).groupBy(
    "Fecha", "Nombre_Estacion"
).agg(
    collect_list("info").alias("info")
)
val df2 = df1.select(
    col("Fecha").alias("_id"), struct(col("Nombre_Estacion"), col("info")).as("details"))
.groupBy("_id").agg(collect_list("details").alias("details"))
```





Procesamiento

```
{
  "id": "2019-04-07 00:00:00",
  "details": [
    {
      "Nombre_Estacion": "Méndez Álvaro",
      "info": [
        {
          "PUNTO_MUESTREO": "28079047_7_8",
          "MAGNITUD_NOMBRE": "Monóxido de Nitrógeno",
          "values": {
            "H01_": "1.0",
            "H02_": "1.0",
            "H03_": "1.0",
            "H04_": "1.0",
            "H05_": "1.0",
            "H06_": "1.0",
            "H07_": "1.0",
            "H08_": "1.0",
            "H09_": "1.0",
            "H10_": "1.0",
            "H11_": "1.0",
            "H12_": "1.0"
          }
        },
        {
          "PUNTO_MUESTREO": "28079047_8_8",
          "MAGNITUD_NOMBRE": "Dióxido de Nitrógeno",
          "values": {
            "H01_": "15.0",
            "H02_": "15.0"
          }
        }
      ]
    }
  ]
}
```





DETALLES DE ALMACENAMIENTO

■ ¿Por qué mongo?

- ▶ Conector con flume y spark.
- ▶ Escalable: si se incluyen más provincias se podría escalar.
- ▶ Tolerante a fallos.
- ▶ Consultas rápidas sobre campos indexados (Fecha, estación, etc),





Almacenamiento

DETALLES DE ALMACENAMIENTO

■ Índices:

- ▷ Se crean índices en los campos:
 - ▷ _id (fecha)
 - ▷ Estación
 - ▷ Magnitud

■ Shard-key:

- ▷ Se eligió el campo "ESTACION" del tipo "hashed" :
 - ▷ Es un campo por el que se realizan muchas de las consultas. El datacenter debería estar preparado para soportar consultas de madrid y luego escalar si se abren nuevas localizaciones.
 - ▷ No es creciente/decreciente.
 - ▷ Si se incluyen más localizaciones, se abre la posibilidad de escalar.





DETALLES DE LA INDEXACIÓN



```
23
24 > db.users.createIndex({"_id":-1})
25 {
26   "createdCollectionAutomatically" : false,
27   "numIndexesBefore" : 1,
28   "numIndexesAfter" : 1,
29   "note" : "all indexes already exist",
30   "ok" : 1
31 }
32 > db.users.createIndex({"details.Nombre_Estacion":1})
33 {
34   "createdCollectionAutomatically" : false,
35   "numIndexesBefore" : 1,
36   "numIndexesAfter" : 2,
37   "ok" : 1
38 }
39 > db.users.createIndex({"details.info.MAGNITUD_NOMBRE":1})
40 {
41   "createdCollectionAutomatically" : false,
42   "numIndexesBefore" : 2,
43   "numIndexesAfter" : 3,
44   "ok" : 1
45 }
```

```
> db.users.getIndexes()
[
  {
    "v" : 1,
    "key" : {
      "_id" : 1
    },
    "name" : "_id_",
    "ns" : "test.users"
  },
  {
    "v" : 1,
    "key" : {
      "details.Nombre_Estacion" : 1
    },
    "name" : "details.Nombre_Estacion_1",
    "ns" : "test.users"
  },
  {
    "v" : 1,
    "key" : {
      "details.info.MAGNITUD_NOMBRE" : 1
    },
    "name" : "details.info.MAGNITUD_NOMBRE_1",
    "ns" : "test.users"
  }
]
```



SOLO MÉNDEZ ÁLVARO

```
> db.users.find({"_id" : /. *20190407.*/}, {details: {$elemMatch :
{"Nombre_Estacion": "Méndez Álvaro"}}}).pretty();
{
  "_id" : "20190407",
  "details" : [
    {
      "Nombre_Estacion" : "Méndez Álvaro",
      "info" : [
        {
          "PUNTO_MUESTREO" : "28079047_7_8",
          "MAGNITUD_NOMBRE" : "Monóxido de Nitrógeno",
          "values" : {
            "H01_" : "V1.0",
            "H02_" : "V1.0",
            "H03_" : "V1.0",
            "H04_" : "V1.0",
            "H05_" : "V1.0",
            "H06_" : "V1.0",
            "H07_" : "V1.0",
            "H08_" : "V1.0",
            "H09_" : "V1.0",
            "H10_" : "V1.0",
            "H11_" : "V1.0",
            "H12_" : "V1.0",
            "H13_" : "V1.0",
            "H14_" : "V1.0",
            "H15_" : "V1.0",
            "H16_" : "V1.0",
            "H17_" : "V1.0",
            "H18_" : "V1.0",
            "H19_" : "V1.0",
            "H20_" : "V1.0",
            "H21_" : "V1.0",
            "H22_" : "V1.0",
            "H23_" : "V1.0",
            "H24_" : "V1.0",
            "H25_" : "V1.0",
            "H26_" : "V1.0",
            "H27_" : "V1.0",
            "H28_" : "V1.0",
            "H29_" : "V1.0",
            "H30_" : "V1.0",
            "H31_" : "V1.0",
            "H32_" : "V1.0",
            "H33_" : "V1.0",
            "H34_" : "V1.0",
            "H35_" : "V1.0",
            "H36_" : "V1.0",
            "H37_" : "V1.0",
            "H38_" : "V1.0",
            "H39_" : "V1.0",
            "H40_" : "V1.0",
            "H41_" : "V1.0",
            "H42_" : "V1.0",
            "H43_" : "V1.0",
            "H44_" : "V1.0",
            "H45_" : "V1.0",
            "H46_" : "V1.0",
            "H47_" : "V1.0",
            "H48_" : "V1.0",
            "H49_" : "V1.0",
            "H50_" : "V1.0",
            "H51_" : "V1.0",
            "H52_" : "V1.0",
            "H53_" : "V1.0",
            "H54_" : "V1.0",
            "H55_" : "V1.0",
            "H56_" : "V1.0",
            "H57_" : "V1.0",
            "H58_" : "V1.0",
            "H59_" : "V1.0",
            "H60_" : "V1.0",
            "H61_" : "V1.0",
            "H62_" : "V1.0",
            "H63_" : "V1.0",
            "H64_" : "V1.0",
            "H65_" : "V1.0",
            "H66_" : "V1.0",
            "H67_" : "V1.0",
            "H68_" : "V1.0",
            "H69_" : "V1.0",
            "H70_" : "V1.0",
            "H71_" : "V1.0",
            "H72_" : "V1.0",
            "H73_" : "V1.0",
            "H74_" : "V1.0",
            "H75_" : "V1.0",
            "H76_" : "V1.0",
            "H77_" : "V1.0",
            "H78_" : "V1.0",
            "H79_" : "V1.0",
            "H80_" : "V1.0",
            "H81_" : "V1.0",
            "H82_" : "V1.0",
            "H83_" : "V1.0",
            "H84_" : "V1.0",
            "H85_" : "V1.0",
            "H86_" : "V1.0",
            "H87_" : "V1.0",
            "H88_" : "V1.0",
            "H89_" : "V1.0",
            "H90_" : "V1.0",
            "H91_" : "V1.0",
            "H92_" : "V1.0",
            "H93_" : "V1.0",
            "H94_" : "V1.0",
            "H95_" : "V1.0",
            "H96_" : "V1.0",
            "H97_" : "V1.0",
            "H98_" : "V1.0",
            "H99_" : "V1.0",
            "H100_" : "V1.0",
            "H101_" : "V1.0",
            "H102_" : "V1.0",
            "H103_" : "V1.0",
            "H104_" : "V1.0",
            "H105_" : "V1.0",
            "H106_" : "V1.0",
            "H107_" : "V1.0",
            "H108_" : "V1.0",
            "H109_" : "V1.0",
            "H110_" : "V1.0",
            "H111_" : "V1.0",
            "H112_" : "V1.0",
            "H113_" : "V1.0",
            "H114_" : "V1.0",
            "H115_" : "V1.0",
            "H116_" : "V1.0",
            "H117_" : "V1.0",
            "H118_" : "V1.0",
            "H119_" : "V1.0",
            "H120_" : "V1.0",
            "H121_" : "V1.0",
            "H122_" : "V1.0",
            "H123_" : "V1.0",
            "H124_" : "V1.0",
            "H125_" : "V1.0",
            "H126_" : "V1.0",
            "H127_" : "V1.0",
            "H128_" : "V1.0",
            "H129_" : "V1.0",
            "H130_" : "V1.0",
            "H131_" : "V1.0",
            "H132_" : "V1.0",
            "H133_" : "V1.0",
            "H134_" : "V1.0",
            "H135_" : "V1.0",
            "H136_" : "V1.0",
            "H137_" : "V1.0",
            "H138_" : "V1.0",
            "H139_" : "V1.0",
            "H140_" : "V1.0",
            "H141_" : "V1.0",
            "H142_" : "V1.0",
            "H143_" : "V1.0",
            "H144_" : "V1.0",
            "H145_" : "V1.0",
            "H146_" : "V1.0",
            "H147_" : "V1.0",
            "H148_" : "V1.0",
            "H149_" : "V1.0",
            "H150_" : "V1.0",
            "H151_" : "V1.0",
            "H152_" : "V1.0",
            "H153_" : "V1.0",
            "H154_" : "V1.0",
            "H155_" : "V1.0",
            "H156_" : "V1.0",
            "H157_" : "V1.0",
            "H158_" : "V1.0",
            "H159_" : "V1.0",
            "H160_" : "V1.0",
            "H161_" : "V1.0",
            "H162_" : "V1.0",
            "H163_" : "V1.0",
            "H164_" : "V1.0",
            "H165_" : "V1.0",
            "H166_" : "V1.0",
            "H167_" : "V1.0",
            "H168_" : "V1.0",
            "H169_" : "V1.0",
            "H170_" : "V1.0",
            "H171_" : "V1.0",
            "H172_" : "V1.0",
            "H173_" : "V1.0",
            "H174_" : "V1.0",
            "H175_" : "V1.0",
            "H176_" : "V1.0",
            "H177_" : "V1.0",
            "H178_" : "V1.0",
            "H179_" : "V1.0",
            "H180_" : "V1.0",
            "H181_" : "V1.0",
            "H182_" : "V1.0",
            "H183_" : "V1.0",
            "H184_" : "V1.0",
            "H185_" : "V1.0",
            "H186_" : "V1.0",
            "H187_" : "V1.0",
            "H188_" : "V1.0",
            "H189_" : "V1.0",
            "H190_" : "V1.0",
            "H191_" : "V1.0",
            "H192_" : "V1.0",
            "H193_" : "V1.0",
            "H194_" : "V1.0",
            "H195_" : "V1.0",
            "H196_" : "V1.0",
            "H197_" : "V1.0",
            "H198_" : "V1.0",
            "H199_" : "V1.0",
            "H200_" : "V1.0",
            "H201_" : "V1.0",
            "H202_" : "V1.0",
            "H203_" : "V1.0",
            "H204_" : "V1.0",
            "H205_" : "V1.0",
            "H206_" : "V1.0",
            "H207_" : "V1.0",
            "H208_" : "V1.0",
            "H209_" : "V1.0",
            "H210_" : "V1.0",
            "H211_" : "V1.0",
            "H212_" : "V1.0",
            "H213_" : "V1.0",
            "H214_" : "V1.0",
            "H215_" : "V1.0",
            "H216_" : "V1.0",
            "H217_" : "V1.0",
            "H218_" : "V1.0",
            "H219_" : "V1.0",
            "H220_" : "V1.0",
            "H221_" : "V1.0",
            "H222_" : "V1.0",
            "H223_" : "V1.0",
            "H224_" : "V1.0",
            "H225_" : "V1.0",
            "H226_" : "V1.0",
            "H227_" : "V1.0",
            "H228_" : "V1.0",
            "H229_" : "V1.0",
            "H230_" : "V1.0",
            "H231_" : "V1.0",
            "H232_" : "V1.0",
            "H233_" : "V1.0",
            "H234_" : "V1.0",
            "H235_" : "V1.0",
            "H236_" : "V1.0",
            "H237_" : "V1.0",
            "H238_" : "V1.0",
            "H239_" : "V1.0",
            "H240_" : "V1.0",
            "H241_" : "V1.0",
            "H242_" : "V1.0",
            "H243_" : "V1.0",
            "H244_" : "V1.0",
            "H245_" : "V1.0",
            "H246_" : "V1.0",
            "H247_" : "V1.0",
            "H248_" : "V1.0",
            "H249_" : "V1.0",
            "H250_" : "V1.0",
            "H251_" : "V1.0",
            "H252_" : "V1.0",
            "H253_" : "V1.0",
            "H254_" : "V1.0",
            "H255_" : "V1.0",
            "H256_" : "V1.0",
            "H257_" : "V1.0",
            "H258_" : "V1.0",
            "H259_" : "V1.0",
            "H260_" : "V1.0",
            "H261_" : "V1.0",
            "H262_" : "V1.0",
            "H263_" : "V1.0",
            "H264_" : "V1.0",
            "H265_" : "V1.0",
            "H266_" : "V1.0",
            "H267_" : "V1.0",
            "H268_" : "V1.0",
            "H269_" : "V1.0",
            "H
```

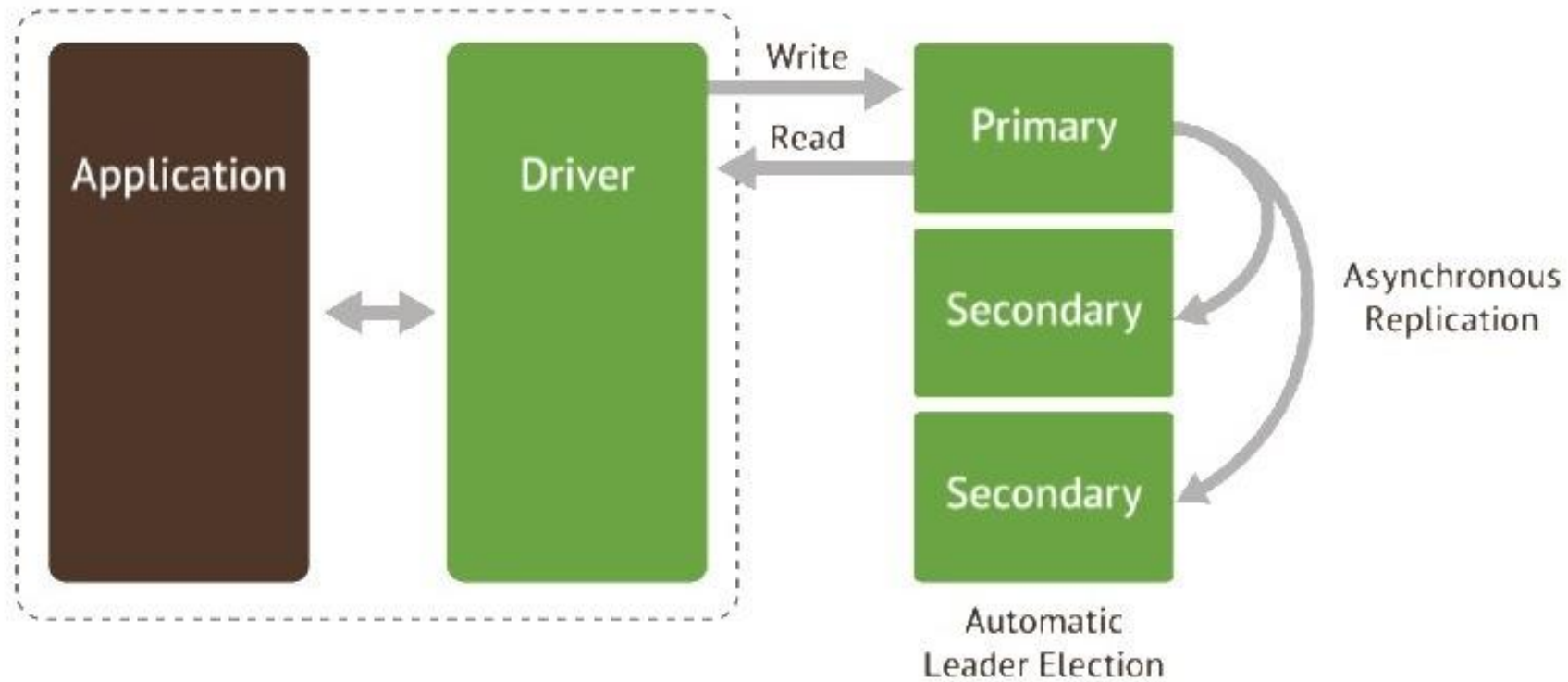
```
"PUNTO_MUESTREO" : "28079047_9_47",
"MAGNITUD_NOMBRE" : "Partículas < 2.5 µm",
"values" : {
  "H01_" : "V4.0",
  "H02_" : "V2.0",
  "H03_" : "V1.0",
  "H04_" : "V1.0",
  "H05_" : "V2.0",
  "H06_" : "V2.0",
  "H07_" : "V3.0",
  "H08_" : "V2.0",
  "H09_" : "V2.0",
  "H10_" : "V1.0",
  "H11_" : "V1.0",
  "H12_" : "V1.0",
  "H13_" : "",
  "H14_" : "",
  "H15_" : "",
  "H16_" : "",
  "H17_" : ""
}
```

```
"PUNTO_MUESTREO" : "28079047_12_8",
"MAGNITUD_NOMBRE" : "Óxidos de Nitrógeno",
"values" : {
  "H01_" : "V17.0",
  "H02_" : "V13.0",
  "H03_" : "V11.0",
  "H04_" : "V8.0",
  "H05_" : "V7.0",
  "H06_" : "V8.0",
  "H07_" : "V10.0",
  "H08_" : "V13.0",
  "H09_" : "V14.0",
  "H10_" : "V12.0",
  "H11_" : "V9.0",
  "H12_" : "V9.0",
  "H13_" : "",
  "H14_" : "",
  "H15_" : "",
  "H16_" : "",
  "H17_" : "",
  "H18_" : "",
  "H19_" : ""
}
```

```
{
  "PUNTO_MUESTREO" : "28079047_8_8",
  "MAGNITUD_NOMBRE" : "Dióxido de Nitrógeno",
  "values" : {
    "H01_" : "V15.0",
    "H02_" : "V12.0",
    "H03_" : "V10.0",
    "H04_" : "V7.0",
    "H05_" : "V6.0",
    "H06_" : "V7.0",
    "H07_" : "V8.0",
    "H08_" : "V11.0",
  }
}
```




DETALLES LA INFRAESTRUCTURA





Almacenamiento – Explotación

DETALLES DE LA CONEXIÓN MONGO- DRILL



```
1  {
2    "type": "mongo",
3    "connection": "mongodb://localhost:27017/",
4    "enabled": true
5  }
6
```



QUERYS

```
1 0: jdbc:drill:zk=local> show databases;
2 +-----+
3 | SCHEMA_NAME |
4 +-----+
5 | cp.default  |
6 | dfs.default |
7 | dfs.root    |
8 | dfs.tmp     |
9 | information_schema |
10 | mongo.db    |
11 | mongo.local |
12 | mongo.test  |
13 | sys         |
14 +-----+
15
16 0: jdbc:drill:zk=local> use mongo.test;
17 +-----+
18 | ok | summary |
19 +-----+
20 | true | Default schema changed to [mongo.test] |
21 +-----+
22 1 row selected (0.077 seconds)
23
24 0: jdbc:drill:zk=local> show tables;
25 +-----+-----+
26 | TABLE_SCHEMA | TABLE_NAME |
27 +-----+-----+
28 | mongo.test    | data        |
29 | mongo.test    | data2       |
30 | mongo.test    | students    |
31 | mongo.test    | del         |
32 | mongo.test    | users       |
33 +-----+-----+
```





Explotación

QUERYS

```
34
35 +-----+
36 |      _id      |      details      |
37 +-----+
38 | 20190407 | [{"Nombre_Estacion":"C/ Farolillo","info":[{"PUNTO_MUESTREO":"28079018_1_38","MAGNITUD_NOMBRE":"Dióxido de Azufre",
39 "values":{"H01_":"V9.0","H02_":"V9.0","H03_":"V9.0","H04_":"V9.0","H05_":"V8.0","H06_":"V8.0","H07_":"V8.0","H08_":"V8.0","H09_":"V8.0",
40 "H10_":"V8.0","H11_":"V9.0","H12_":"V8.0","H13_":"","H14_":"","H15_":"","H16_":"","H17_":"","H18_":"","H19_":"","H20_":"","H21_":"","H22_":
41 "","H23_":"","H24_":"}},{PUNTO_MUESTREO":"28079018_6_48","MAGNITUD_NOMBRE":"Monóxido de Carbono","values":{"H01_":"V0.3","H02_":"V0.3",
42 "H03_":"V0.3","H04_":"V0.3","H05_":"V0.3","H06_":"V0.3","H07_":"V0.3","H08_":"V0.3","H09_":"V0.3","H10_":"V0.3","H11_":"V0.3","H12_":"V0.3","H13_":
43
44
45 0: jdbc:drill:zk=local> select d._id, d.Estaciones, d.P_MUESTREO, d.MAGNITUD
46 . . . . . semicolon>         from (select b._id, b.Estaciones, b.info.PUNTO_MUESTREO P_MUESTREO, b.info.MAGNITUD_NOMBRE MAGNITUD
47 . . . . . )>                 from (select a._id, a.det.Nombre_Estacion Estaciones, flatten(a.det.info) info
48 . . . . . )>                 from (select u._id, flatten(u.details) det from users u) a)b)_id
49 . . . . . semicolon> ;
50 +-----+
51 |      _id      |      Estaciones      |      P_MUESTREO      |      MAGNITUD      |
52 +-----+
53 | 20190407 | C/ Farolillo | 28079018_1_38 | Dióxido de Azufre |
54 | 20190407 | C/ Farolillo | 28079018_6_48 | Monóxido de Carbono |
55 | 20190407 | C/ Farolillo | 28079018_7_8 | Monóxido de Nitrógeno |
56 | 20190407 | C/ Farolillo | 28079018_8_8 | Dióxido de Nitrógeno |
57 | 20190407 | C/ Farolillo | 28079018_10_47 | Partículas < 10 µm |
58 | 20190407 | C/ Farolillo | 28079018_12_8 | Óxidos de Nitrógeno |
59 | 20190407 | C/ Farolillo | 28079018_14_6 | Ozono |
60 | 20190407 | C/ Farolillo | 28079018_20_59 | Tolueno |
61 | 20190407 | C/ Farolillo | 28079018_30_59 | Benceno |
62 | 20190407 | C/ Farolillo | 28079018_35_59 | Etilbenceno |
63 | 20190407 | Plaza Elíptica | 28079056_6_48 | Monóxido de Carbono |
64 | 20190407 | Plaza Elíptica | 28079056_7_8 | Monóxido de Nitrógeno |
65
```





Explotación



QUERYS

```
0: jdbc:drill:zk=local> create temporary table users_inf as select a._id, a.det.Nombre_Estacion Estaciones, flatten (a.det.info) info
. . . . .semicolon> from (select u._id, flatten(u.details) det from users u) a;
```

Fragment	Number of records written
0_0	151

```
0: jdbc:drill:zk=local> select b._id, b.Estaciones, b.P_MUESTREO, b.MAGNITUD, b.val.H01_, b.val.H02_, b.val.H03_, b.val.H04_, b.val.H05_, b.val.H06_, b.val.H08_, b.val.H09_, b.val.H10_,
. . . . .semicolon> b.val.H11_, b.val.H12_, b.val.H13_, b.val.H14_, b.val.H15_, b.val.H16_, b.val.H17_, b.val.H18_, b.val.H19_, b.val.H20_, b.val.H21_, b.val.H22_, b.val.H23_, b.val.H24_
. . . . .semicolon> from (select i._id, i.Estaciones, i.info.PUNTO_MUESTREO P_MUESTREO, i.info.MAGNITUD_NOMBRE MAGNITUD, flatten (i.info.values) val
. . . . .)>
from users_inf i;
```

_id	Estaciones	P_MUESTREO	MAGNITUD	H01_	H02_	H03_	H04_	H05_	H06_	H07_	H08_	H09_	H10_	H11_	H12_	H13_	H14_	H15_	H16_	H17_	H18_	H19_	H20_	H21_	H22_	H23_	H24_
20190407	C/ Farolillo	28079018_1_38	Dióxido de Azufre	null	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	null	null	null	null	null	null	null	null	null	null	null
20190407	C/ Farolillo	28079018_6_48	Monóxido de Carbono	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	null	null	null	null	null	null	null	null	null	null	null
20190407	C/ Farolillo	28079018_7_8	Monóxido de Nitrógeno	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0
20190407	C/ Farolillo	28079018_8_8	Dióxido de Nitrógeno	19.0	19.0	19.0	19.0	19.0	19.0	19.0	19.0	19.0	19.0	19.0	19.0	19.0	19.0	null	null	null	null	null	null	null	null	null	null
20190407	C/ Farolillo	28079018_10_47	Partículas < 10 µm	27.0	27.0	27.0	27.0	27.0	27.0	27.0	27.0	27.0	27.0	27.0	27.0	27.0	null	null	null	null	null	null	null	null	null	null	null
20190407	C/ Farolillo	28079018_12_8	Óxidos de Nitrógeno	8.0	8.0	8.0	8.0	8.0	8.0	8.0	8.0	8.0	8.0	8.0	8.0	8.0	null	null	null	null	null	null	null	null	null	null	null
20190407	C/ Farolillo	28079018_14_6	Ozono	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	null	null	null	null	null	null	null	null	null	null	null
20190407	C/ Farolillo	28079018_20_59	Tolueno	8.0	8.0	8.0	8.0	8.0	8.0	8.0	8.0	8.0	8.0	8.0	8.0	8.0	null	null	null	null	null	null	null	null	null	null	null

A photograph of Donald Trump, wearing a dark suit, white shirt, and a blue and white striped tie. He is looking down with a pained or distressed expression. In the foreground, two young children are crying. The child on the left is a girl with a pink bow in her hair, wearing a purple patterned top. The child on the right is a boy with blonde hair, wearing a blue patterned shirt. The background is blurred, showing other people. A semi-transparent blue rectangle is overlaid on the image, containing the text.

Problemas y aspectos a mejorar



PROBLEMAS Y ASPECTOS A MEJORAR

Problemas:

- Generación de json anidados.
- Problemas para acceder a los Arrays con subjsons porque por mucho que replicabamos los pasos de la documentación, no nos funcionaba.
- Consultas de json con Drill.

Aspectos a mejorar:

- Poder implementar el indicador del aire debido a que es una función compleja.



Explotación

KPI: ICA (Indicador de Calidad del Aire)

Sistema para informar sobre la gravedad de los niveles de calidad del aire, en términos entendibles con un resultado numérico que varía de 1 a 150, siendo el valor más alto el de mayores consecuencias.

CONTAMINANTE	VALOR LÍMITE
Dióxido de Azufre (SO ₂)	VL horario= 125 µg/m ³
Dióxido de Nitrógeno (NO ₂)	VL horario= 200 µg/m ³
Partículas (PM ₁₀)	VL horario= 50 µg/m ³
Ozono (O ₃)	VL horario=180 µg/m ³
Monóxido de carbono (CO)	VL horario=10 µg/m ³

Índice	Calidad del Aire	Color Descriptivo
≥ 0 y < 75	Buena	● Verde
≥ 75 y < 100	Admisible	● Amarillo
≥ 100 y < 150	Mala	● Rojo
≥ 150	Muy Mala	● Violeta

¡Muchas gracias!

Alexander Balseiro
Brian Ferrol
Álvaro Cerrato
Ángel Rivas