Práctica 5: Introducción a los formatos Avro y Parquet

Para realizar esta práctica son necesarios los archivos **paises.avro** y **demo.parquet**

Uso de archivos Avro

Avro se basa en esquemas. Cuando los datos *.avro* son leídos siempre está presente el esquema con el que han sido escritos. Esto permite aumentar el rendimiento al escribir los datos, haciendo la serialización rápida y viable en espacio.

Para poder analizar, visualizar sus datos y transformar los ficheros *avro* vamos a utilizar las herramientas **avro-tools**

Esquema de un fichero avro

Conseguimos el formato de los datos de un fichero avro a través de su esquema. Se ejecuta la siguiente instrucción:

```
avro-tools getschema paises.avro > paises.avsc
```

```
cat  paises.avsc
```

```
[alumno@pasarela avro-parquet]$ cat paises.avsc
{
  "type" : "record",
  "name" : "Root",
  "fields" : [ {
    "name" : "country",
    "type" : [ "null", "string" ]
  }, {
    "name" : "year",
    "type" : [ "null", "long" ]
  }, {
    "name" : "population",
    "type" : [ "null", "double" ]
  }, {
    "name" : "continent",
    "type" : [ "null", "string" ]
  }, {
    "name" : "lifeExp",
    "type" : [ "null", "double" ]
  }, {
    "name" : "gdpPercap",
    "type" : [ "null", "double" ]
  } ]
}
[alumno@pasarela avro-parquet]$
```

Figure 1: Esquema json de fichero avro

Extracción de datos desde un fichero avro

Con las **avro-tools** se pueden obtener los datos de un fichero avro en distintos formatos. En este caso, el esquema y datos del fichero vienen en formato json.

Exportar datos a formato JSON Utilizamos la opción *tojson* (redireccionando la salida a un fichero *.json*)

```
avro-tools tojson paises.avro > paises.json
```

Y mostramos las primeras líneas del fichero *paises.json*

```
[alumno@pasarela avro-parquet]$ avro-tools tojson paises.avro > paises.json
[main] WARN org.apache.hadoop.util.NativeCodeLoader - Unable to load native-hadoop library for your platform... using builtin-java classes where
applicable
log4j:WARN No appenders could be found for logger (org.apache.htrace.core.Tracer).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
[alumno@pasarela avro-parquet]$ ls
demo.parquet  distribuidores.parquet  paises.avro  paises.avsc  paises.json
[alumno@pasarela avro-parquet]$ head paises.json
{"country":{"string":"Afghanistan"},"year":{"long":1952},"population":{"double":8425333.0},"continent":{"string":"Asia"},"lifeExp":{"double":"NaN
"},"gdpPercap":{"double":779.4453145}}
{"country":{"string":"Afghanistan"},"year":{"long":1957},"population":{"double":9240934.0},"continent":{"string":"Asia"},"lifeExp":{"double":30.3
31999999999997},"gdpPercap":{"double":820.8530296}}
{"country":{"string":"Afghanistan"},"year":{"long":1962},"population":{"double":1.0267083E7},"continent":{"string":"Asia"},"lifeExp":{"double":31
.997},"gdpPercap":{"double":853.1007099999999}}
{"country":{"string":"Afghanistan"},"year":{"long":1967},"population":{"double":1.1537966E7},"continent":{"string":"Asia"},"lifeExp":{"double":34
.02},"gdpPercap":{"double":836.1971382}}
{"country":{"string":"Afghanistan"},"year":{"long":1972},"population":{"double":1.307946E7},"continent":{"string":"Asia"},"lifeExp":{"double":36.
088},"gdpPercap":{"double":739.9811057999999}}
{"country":{"string":"Afghanistan"},"year":{"long":1977},"population":{"double":1.4880372E7},"continent":{"string":"Asia"},"lifeExp":{"double":38
.438},"gdpPercap":{"double":786.11336}}
```

Figure 2: Líneas json desde un fichero avro

Uso de ficheros Parquet

Es un formato de almacenamiento columnar disponible para cualquier proyecto en el ecosistema de Hadoop, independiente del framework utilizado para procesar los datos, o el lenguaje de programación.

En este caso utilizamos la aplicación **parquet-tools**, y el fichero de muestra es *demo.parquet*

Visualización de los datos del archivo parquet

Se pueden obtener las líneas del archivo en texto plano, usando la opción **cat**, y a partir de la salida procesar el fichero obtenido.

```
[alumno@pasarela avro-parquet]$ parquet-tools cat demo.parquet > demo.txt
[alumno@pasarela avro-parquet]$ head demo.txt
id = 1
name = Toy Story
year = 1995

id = 2
name = Jumanji
year = 1995

id = 3
name = Grumpier Old Men
```

Figure 3: Exportar a texto un fichero parquet

O bien visualizar directamente las primeras líneas con la opción **head**

```
parquet-tools head demo.parquet
```

```
[alumno@pasarela avro-parquet]$ parquet-tools head demo.parquet
id = 1
name = Toy Story
year = 1995

id = 2
name = Jumanji
year = 1995

id = 3
name = Grumpier Old Men
year = 1995

id = 4
name = Waiting to Exhale
year = 1995

id = 5
name = Father of the Bride Part II
year = 1995
```

Figure 4: Exportar a texto un fichero parquet

Obtención del esquema de los datos

parquet-tools permite obtener el esquema de la estructura de los datos de forma simple, o bien de forma detallada usando la opción *-d*

`parquet-tools schema demo.parquet`

```
[alumno@pasarela avro-parquet]$ parquet-tools schema  demo.parquet
message movie {
  optional int32 id;
  optional binary name (UTF8);
  optional int32 year;
}

[alumno@pasarela avro-parquet]$ parquet-tools schema -d demo.parquet
message movie {
  optional int32 id;
  optional binary name (UTF8);
  optional int32 year;
}

creator: parquet-mr version 1.9.0-cdh6.1.1 (build ${buildNumber})
extra: parquet.avro.schema = {"type":"record","name":"movie","doc":"Sqoop import of movie","fields":[{"name":"id","type":["null","int"],"default"
:null,"columnName":"id","sqlType":"4"},{"name":"name","type":["null","string"],"default":null,"columnName":"name","sqlType":"1"},{"name":"year","
type":["null","int"],"default":null,"columnName":"year","sqlType":"5"}],"tableName":"movie"}
extra: writer.model.name = avro

file schema: movie
--------------------------------------------------------------------------------------------------------------------------------------------------------
id: OPTIONAL INT32 R:0 D:1
name: OPTIONAL BINARY O:UTF8 R:0 D:1
year: OPTIONAL INT32 R:0 D:1

row group 1: RC:3881 TS:96157 OFFSET:4
--------------------------------------------------------------------------------------------------------------------------------------------------------
id:  INT32 SNAPPY DO:0 FPO:4 SZ:15585/15581/1,00 VC:3881 ENC:BIT_PACKED,RLE,PLAIN ST:[min: 1, max: 3952, num_nulls: 0]
name:  BINARY SNAPPY DO:0 FPO:15589 SZ:53871/76846/1,43 VC:3881 ENC:BIT_PACKED,RLE,PLAIN ST:[min: $1,000,000 Duck, max: eXistenZ, num_nulls: 0]
year:  INT32 SNAPPY DO:0 FPO:69460 SZ:3554/3730/1,05 VC:3881 ENC:BIT_PACKED,RLE,PLAIN_DICTIONARY ST:[min: 0, max: 2000, num_nulls: 0]
```

Figure 5: Esquema de un fichero parquet

Obtención de los datos de una columna

Con la opción **dump** se pueden extraer los datos y metadatos de una o varias columnas.

```
[alumno@pasarela avro-parquet]$ parquet-tools dump -c name demo.parquet > demo.dump.out
[alumno@pasarela avro-parquet]$ head -20 demo.dump.out
row group 0
-------------------------------------------------------------------------------------------------------
name:  BINARY SNAPPY DO:0 FPO:15589 SZ:53871/76846/1,43 VC:3881 ENC:RLE,BIT_PACKED,PLAIN ST:[min: $1,000,000 Duck, max: eXistenZ, num_nulls: 0]

    name TV=3881 RL=0 DL=1
    -------------------------------------------------------------------------------------------------
    page 0:  DLE:RLE RLE:BIT_PACKED VLE:PLAIN ST:[min: $1,000,000 Duck, max: eXistenZ, num_nulls: 0] SZ:76793 VC:3881

BINARY name
-------------------------------------------------------------------------------------------------------
*** row group 1 of 1, values 1 to 3881 ***
value 1:    R:0 D:1 V:Toy Story
value 2:    R:0 D:1 V:Jumanji
value 3:    R:0 D:1 V:Grumpier Old Men
value 4:    R:0 D:1 V:Waiting to Exhale
value 5:    R:0 D:1 V:Father of the Bride Part II
value 6:    R:0 D:1 V:Heat
value 7:    R:0 D:1 V:Sabrina
value 8:    R:0 D:1 V:Tom and Huck
value 9:    R:0 D:1 V:Sudden Death
```

Figure 6: Selección de una columna de datos