

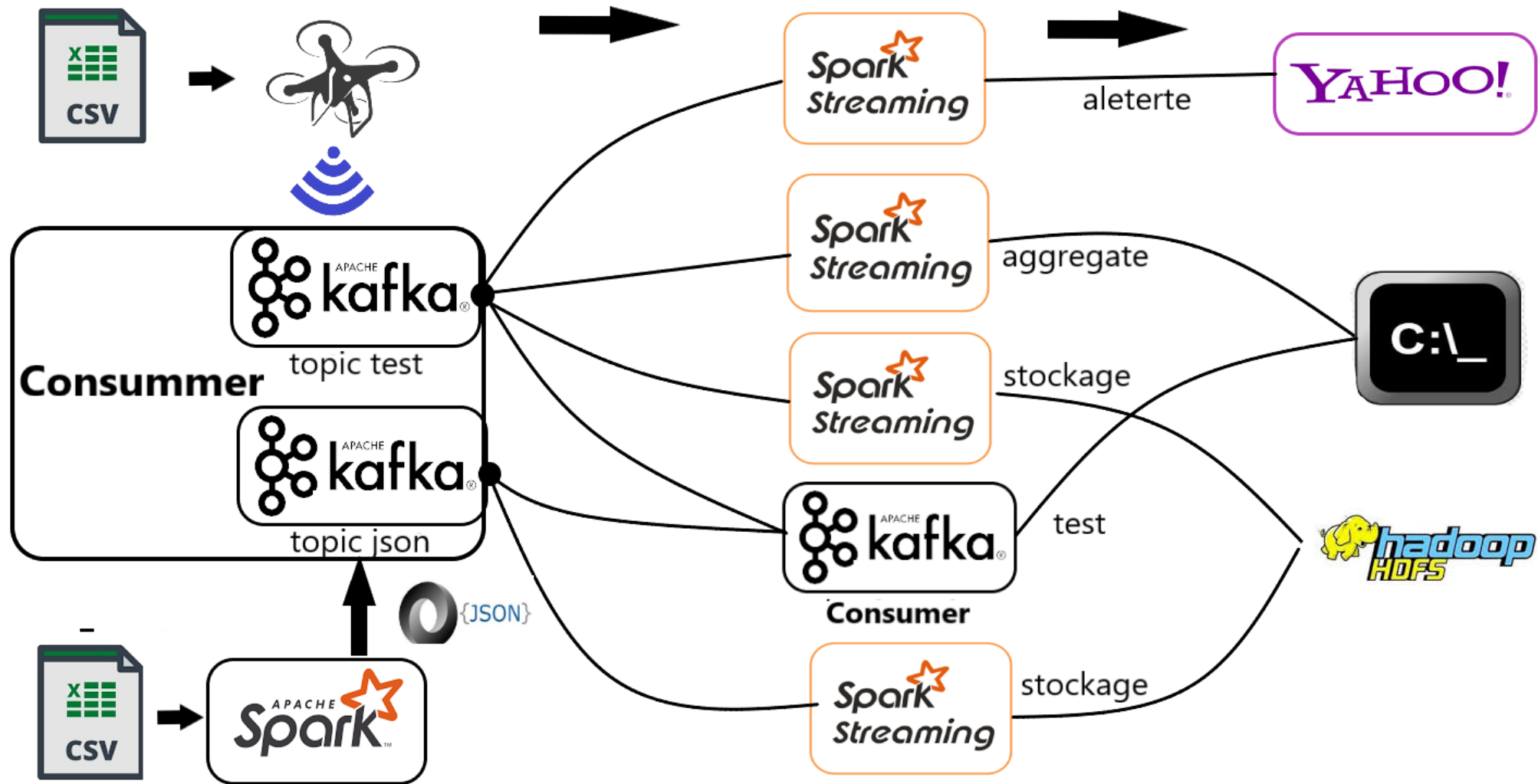
SPARK Project

Prestacop

Présentation du sujet

- Prestacop propose un service de drone pour aider les systèmes de police dans leur mission de contraventions
- Chaque drone envoie un message régulièrement.
- Les messages de chaque drone doivent être stockés.
- Si un drone ne peut pas qualifier un délit, alors il envoie une alerte qui indique qu'il nécessite une intervention humaine.

Architecture



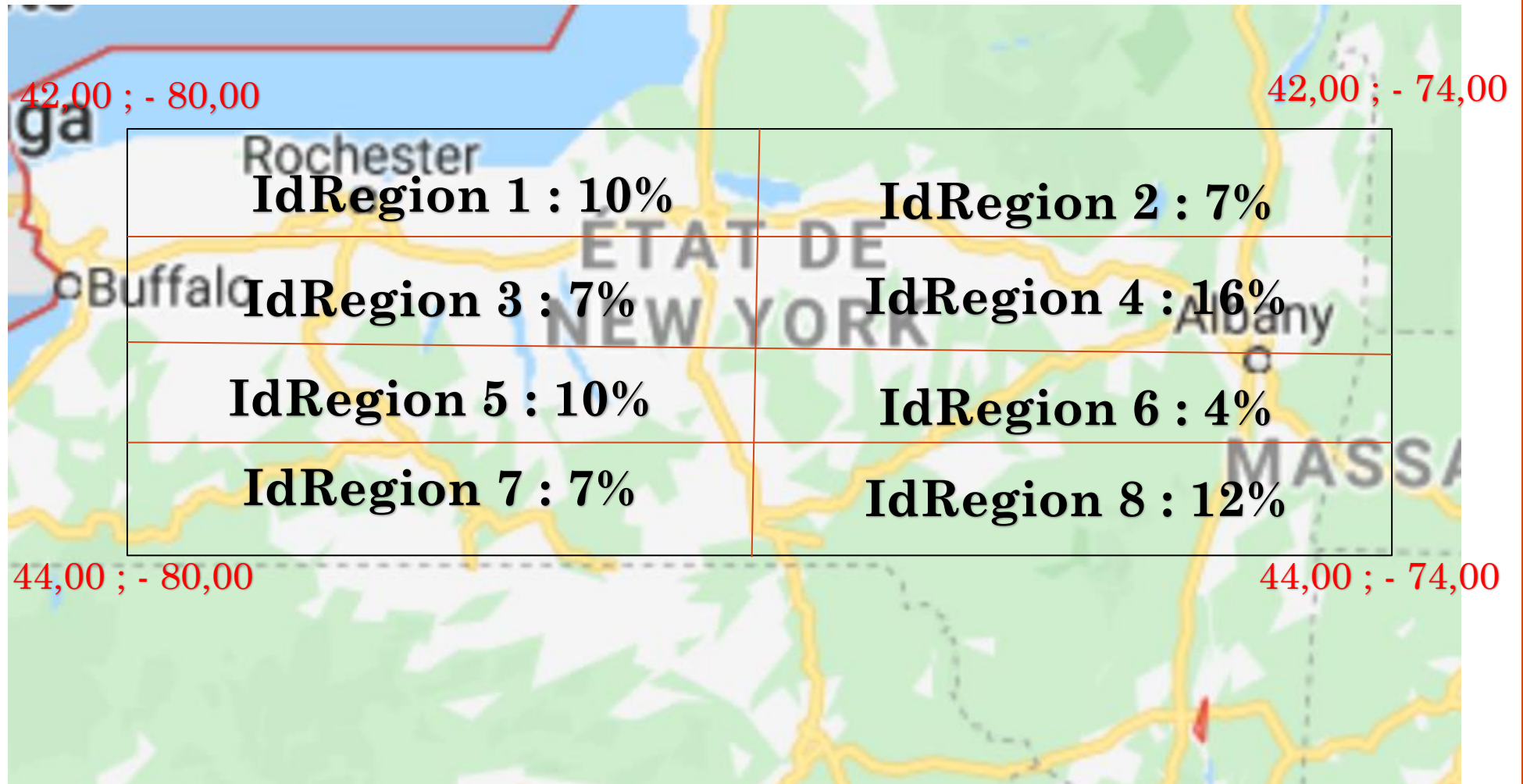
Drone (Producer)

- 56 drones en activité
- Numérotés en fonction de la region : entre 10 et 80 zones
- Envoient des données cohérentes
- Connectés à Kafka (localhost:192.168.99.100, port:9092)
- Topic “test”

Message

- Message => idRegion; idDrone; currentDate; currentTime; x; y;
RegionState; StreetCode1; StreetCode2; StreetCode3;
HouseNumber; StreetName; **Violation** ; idPhoto; plaque; PlateType;
VehicleBodyType; VehicleMake; VehicleColor; violationCode;
violationDescription
- Généré de manière cohérente aléatoirement
- Violation
 - 0 Infraction défini
 - 1 Infraction non défini, intervention humaine requise
 - 2 Aucune infraction

Cartographie des drones



Spark Streaming : Aggregate

- Connecté à Kafka, Topic : test
- Consomme les messages du drone
- Message et calcule en temps réel affiché sur le terminale

```
-----  
Batch: 0  
-----  
+-----+-----+-----+  
|          window|IdDrone|count|  
+-----+-----+-----+  
|[2020-07-15 21:26...|    12|    2|  
|[2020-07-15 21:25...|    42|    4|  
-----
```

```
Batch: 1  
-----  
+-----+-----+-----+  
|          window|IdDrone|count|  
+-----+-----+-----+  
|[2020-07-15 21:26...|    12|    2|  
|[2020-07-15 21:25...|    42|    4|  
-----
```

```
-----  
Batch: 0  
-----  
+-----+-----+-----+  
|          window|VehicleBodyType|count|  
+-----+-----+-----+  
|[2020-07-15 21:25...|      PICK|    2|  
|[2020-07-15 21:25...|      SDN|    4|  
-----
```

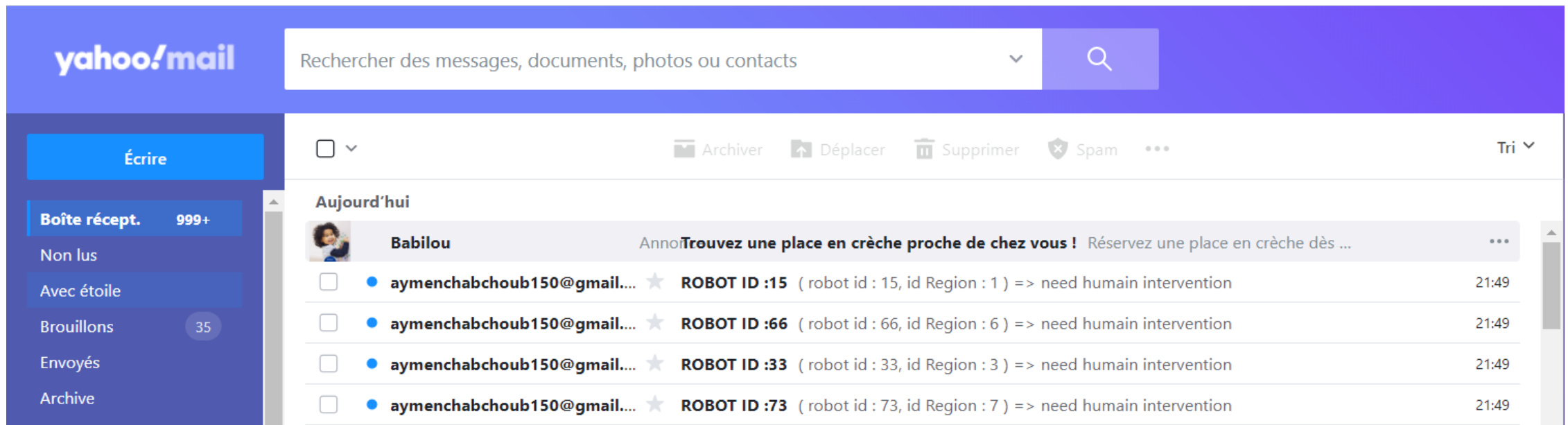
```
Batch: 1  
-----  
+-----+-----+-----+  
|          window|VehicleBodyType|count|  
+-----+-----+-----+  
|[2020-07-15 21:25...|      PICK|    2|  
|[2020-07-15 21:27...|      SUBN|   20|  
-----
```

```
-----  
Batch: 0  
-----  
+-----+-----+-----+  
|          window|ViolationCode|count|  
+-----+-----+-----+  
|[2020-07-15 21:25...|      80|    1|  
|[2020-07-15 21:24...|      35|    1|  
-----
```

```
Batch: 1  
-----  
+-----+-----+-----+  
|          window|ViolationCode|count|  
+-----+-----+-----+  
|[2020-07-15 21:25...|      80|    1|  
|[2020-07-15 21:24...|      35|    1|  
-----
```









Spark Streaming : Alerte

- Connecté a Kafka, Topic : Test
- Connecté au compte Google administrateur
- Envoie un email en temp réel avec id, région du drone nécessitant une intervention humaine (violation = 1)
- Appel de la fonction SendMailer. (file : Mailer.scala)



Spark Streaming : HDFS

- Connecté à Kafka, Topic : Test
- Connecté a Hadoop
- Consomme les messages du drone et les stockes sur HDFS / Local
- Données stockées toutes les 1 minutes en local, toutes les heures en HDFS
- problème de connexion à Apache -> Docker Toolbox host : 192.168.99.100

	_spark_metadata	15/07/2020 22:34	Dossier de fichiers	
	.part-00000-1237f40c-c617-416d-b362-7...	15/07/2020 22:34	Fichier CRC	1 Ko
	.part-00000-3751fe8c-07b1-42e6-9c37-6...	15/07/2020 22:32	Fichier CRC	1 Ko
	.part-00000-b83a88a8-0bc8-4d6e-a39e-...	15/07/2020 22:33	Fichier CRC	1 Ko
	.part-00000-cc12c5aa-4588-4057-99b2-f...	15/07/2020 22:28	Fichier CRC	1 Ko
	part-00000-1237f40c-c617-416d-b362-7...	15/07/2020 22:34	Fichier CSV Micros...	58 Ko
	part-00000-3751fe8c-07b1-42e6-9c37-6...	15/07/2020 22:32	Fichier CSV Micros...	15 Ko
	part-00000-b83a88a8-0bc8-4d6e-a39e-4...	15/07/2020 22:33	Fichier CSV Micros...	93 Ko

Transfert données historiques

- Read csv -> convert Json -> Producer Kafka
- On ne garde que les champs envoyés par le drone
- Json = { IdDrone : Int,
 zone : {var idRegion : Int, RegionState : String, StreetCode1 : String,
 StreetCode2 : String, StreetCode3 : String, HouseNumber : String,
 StreetName : String },
 coordonnee: { longitude: Float, latitude : Float },
 currentDate : String,
 currentTime : String,
 violationType: Int,
 infraction : { code: String, violationDescription : String, idPhoto: String }
 vehicle : { plaque : String, PlateType : String, VehicleBodyType : String,
 VehicleMake : String, VehicleColor : String }
}

Transfert données historiques

- Class Java

```
case class Message( var idDrone : Int , var zone : Zone, var coordonnée : Coordonnee, var currentDate : String, var  
currentTime : String, var violationType: Int, var infraction: Infraction, var vehicle :Vehicle)
```

```
case class Zone(var idRegion : Int,  
var RegionState : String,  
var StreetCode1 : String,  
var StreetCode2 : String,  
var StreetCode3 : String,  
var HouseNumber : String,  
var StreetName : String )
```

```
case class Infraction(var code: String,  
var violationDescription : String,  
var idPhoto: String)
```

```
case class Coordonnee(var longitude: Float,  
var latitude: Float)
```

```
case class Vehicle( var plaque : String,  
var PlateType : String,  
var VehicleBodyType : String,  
var VehicleMake : String,  
var VehicleColor : String)
```

- Envoie d'un élément Message au Producer

Transfert données historiques








- Connecté à Kafka, Topic : Json
- Connecté a Hadoop
- Création d'un schéma Spark au format de Message()

```
schema = (new StructType()  
  .add("idDrone",IntegerType)  
  .add("zone", (new StructType()) .add("idRegion",IntegerType) .add("RegionState",StringType) add("StreetCode1",StringType) add("StreetCode2",StringTring  
  .add("StreetCode3",StringType .add("HouseNumber",StringType) .add("StreetName",StringType)  
  )  
  .add("coordonnée",(new StructType()).add("longitude",FloatType).add("latitude",FloatType)  
  )  
  .add("currentDate",StringType)  
  .add("currentTime",StringType)  
  .add("violationType",IntegerType)  
  .add("infraction",(new StructType()) .add("code",StringType) .add("violationDescription",StringType)  
  . )  
  .add("vehicle",(new StructType()) .add("plaque",StringType) .add("plateType",StringType) .add("VehicleBodyType",StringType)  
  .add("VehicleMake",StringType) .add("VehicleColor",StringType))  
  )
```

- Consomme les messages du KafKa au format Json et les parses

Transfert données historiques

- Connecté à Kafka
- Connecté a Hadoop
- Consomme les messages du drone et les stockes sur HDFS / Local
- Données stockées toutes les 1 minutes en local, toutes les heures en HDFS
- problème de connexion à Apache -> Docker Toolbox host : 192.168.99.100

 _spark_metadata	12/07/2020 18:44	Dossier de fichiers	
 .part-00000-7a8bca38-4d38-4a59-88d1-f...	12/07/2020 18:43	Fichier CRC	139 Ko
 .part-00000-a43c82a4-0c5e-4fca-9200-0...	12/07/2020 18:40	Fichier CRC	1 Ko
 .part-00000-c82b16f2-8af3-455a-bc0b-3...	12/07/2020 18:44	Fichier CRC	148 Ko
 part-00000-7a8bca38-4d38-4a59-88d1-f...	12/07/2020 18:43	Fichier CSV Micros...	17 714 Ko
 part-00000-a43c82a4-0c5e-4fca-9200-09...	12/07/2020 18:40	Fichier CSV Micros...	0 Ko
 part-00000-c82b16f2-8af3-455a-bc0b-3...	12/07/2020 18:44	Fichier CSV Micros...	18 910 Ko