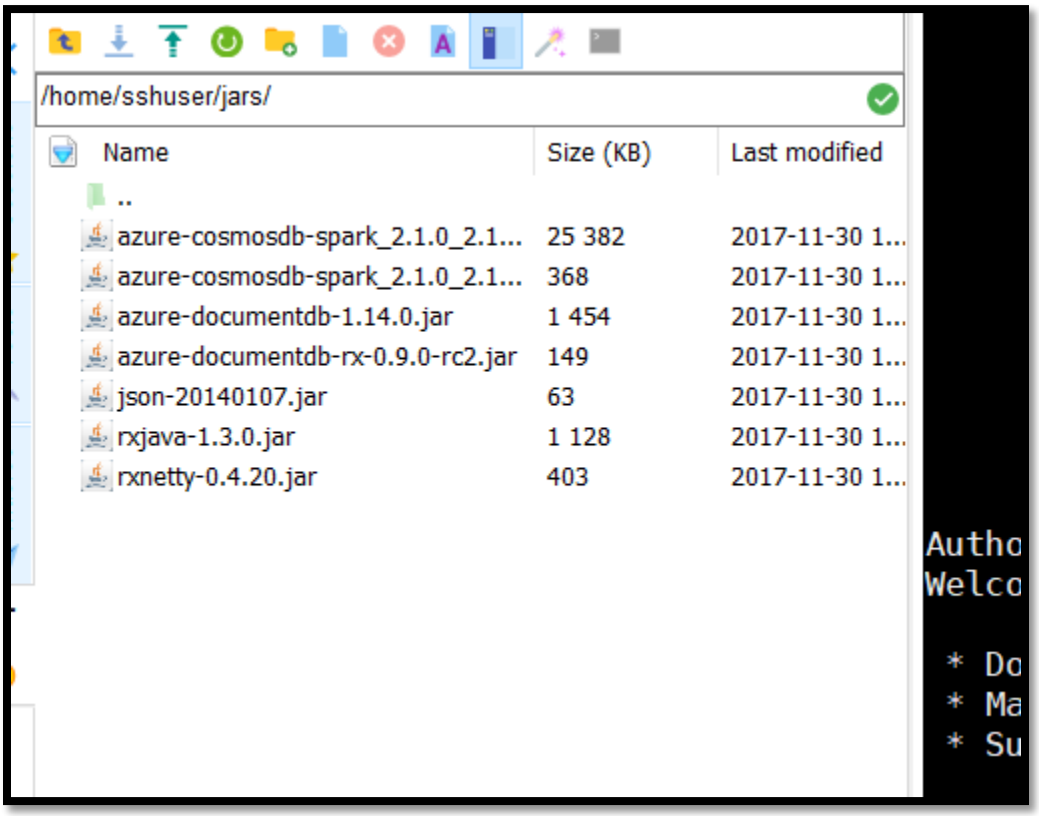
SPARK TO COSMOSDB

Setup Instruction

1) Uploading jar files to the cluster
2) Importing necessary jar files
3) Reading Data from CosmosDB
4) Writing Data to CosmosDB

Uploading jar files to the cluster

1) Download jar files from Maven repository https://mvnrepository.com/artifact/com.microsoft.azure/azure-cosmosdb-spark_2.2.0_2.11/1.0.0
2) Upload jar files to the cluster



Reading Data from COSMOSDB

1) SSH to the cluster
2) Run Following command. This will initialize spark framework with reference of jar files mentioned

spark-shell --master yarn --jars /home/sshuser/jars/azure-cosmosdb-spark_2.1.0_2.11-1.0.0.jar,/home/sshuser/jars/azure-documentdb-1.14.0.jar,/home/sshuser/jars/rxjava-1.3.0.jar,/home/sshuser/jars/azure-documentdb-rx-0.9.0-rc2.jar,/home/sshuser/jars/json-20140107.jar

```
sshuser@hn0-sudhir:~$ spark-shell --master yarn --jars /home/sshuser/jars/azure-cosmosdb-
documentdb-1.14.0.jar,/home/sshuser/jars/rxjava-1.3.0.jar,/home/sshuser/jars/azure-docume
07.jar
SPARK_MAJOR_VERSION is set to 2, using Spark2
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLeve
Spark context Web UI available at http://10.0.0.19:4040
Spark context available as 'sc' (master = yarn, app id = application_1512025053676_0004).
Spark session available as 'spark'.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 2.1.1.2.6.2.3-1
      /_/

Using Scala version 2.11.8 (OpenJDK 64-Bit Server VM, Java 1.8.0_151)
Type in expressions to have them evaluated.
Type :help for more information.

scala>
```

3) Run following command to import reference and add configuration

```
import com.microsoft.azure.cosmosdb.spark.schema._
import com.microsoft.azure.cosmosdb.spark._
import com.microsoft.azure.cosmosdb.spark.config.Config

val ReadConfig = Config(Map("Endpoint" ->
"https://sudhirawdemocdb.documents.azure.com:443/",
"Masterkey" ->
"MstsCKBMXIH8zGfWx4YEhCUcTvBVor9yDYADpg7G393oqYHQX7erWFKE353cWsuoqw3m9nZnx
yjtvQ73lIpmeg==",
"Database" -> "startcosmos",
"preferredRegions" -> "Central India;",
"Collection" -> "assestdata",
"SamplingRatio" -> "1.0"))
```

```
scala> import com.microsoft.azure.cosmosdb.spark.schema._
import com.microsoft.azure.cosmosdb.spark.schema._

scala> import com.microsoft.azure.cosmosdb.spark._
import com.microsoft.azure.cosmosdb.spark._

scala> import com.microsoft.azure.cosmosdb.spark.config.Config
import com.microsoft.azure.cosmosdb.spark.config.Config

scala> val ReadConfig = Config(Map("Endpoint" -> "https://sudhirawdemocdb.documents.azure.com:443/",
     | "Masterkey" -> "MstsCKBMXIH8zGfWx4YEhCUcTvBVor9yDYADpg7G393oqYHQX7erWFKE353cWsuoqw3m9nZnxyjtvQ73lIpmeg==",
     | "Database" -> "startcosmos",
     | "preferredRegions" -> "Central India;",
     | "Collection" -> "assestdata",
     | "SamplingRatio" -> "1.0"))
ReadConfig: com.microsoft.azure.cosmosdb.spark.config.Config = com.microsoft.azure.cosmosdb.spark.config.ConfigBuilder$$anon$1@45ec2e46

scala>
```

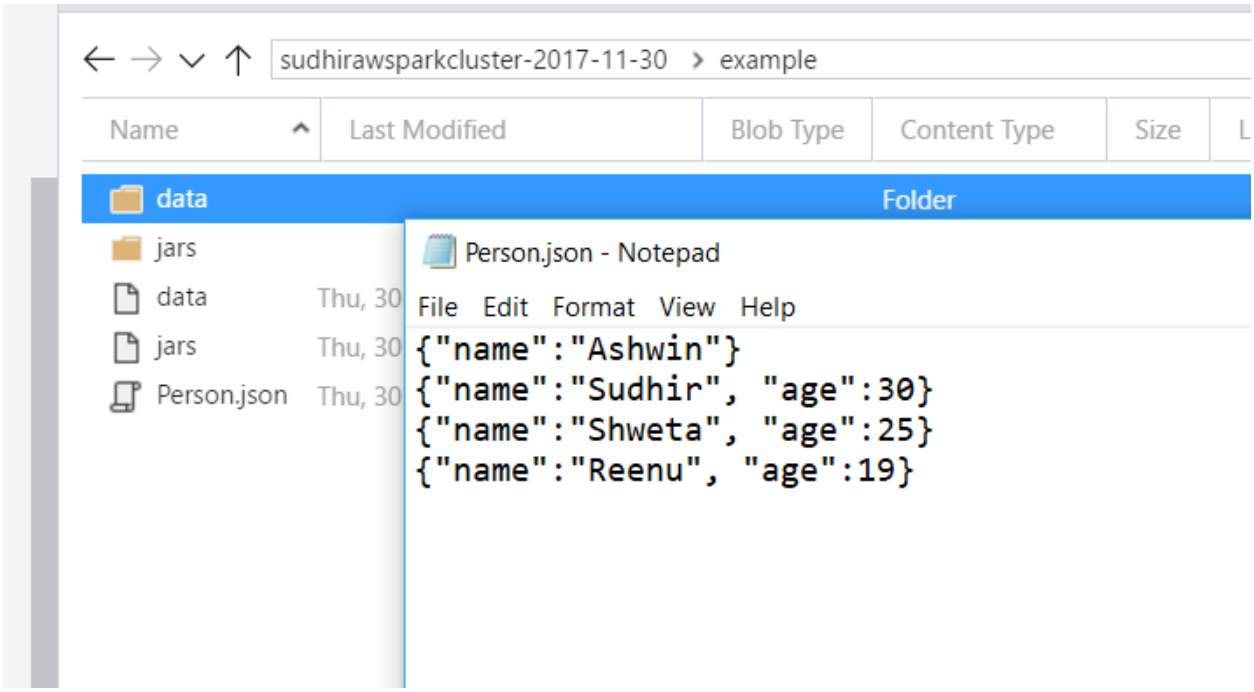
4) Run following command to read data from the collection mentioned in configuration file

```
val coll = spark.sqlContext.read.cosmosDB(ReadConfig)
coll.createOrReplaceTempView("c")
val sqlDF = spark.sql("SELECT *  FROM c ")
sqlDF.show(false)
```

```
scala> val coll = spark.sqlContext.read.cosmosDB(ReadConfig)
17/11/30 13:43:00 WARN CosmosDBConnection: CosmosDBConnection::Input preferred region list: Central India;
17/11/30 13:43:01 WARN ServiceJNIWrapper: 'Linux' with 'amd64' system is not compatible with native library. JNI not loaded.
coll: org.apache.spark.sql.DataFrame = [AlarmWord: string, AssetCode: string ... 36 more fields]

scala> coll.createOrReplaceTempView("c")

scala> val sqlDF = spark.sql("SELECT *  FROM c ")
sqlDF: org.apache.spark.sql.DataFrame = [AlarmWord: string, AssetCode: string ... 36 more fields]

scala> sqlDF.show(false)
17/11/30 13:43:10 WARN Utils: Truncated the string representation of a plan since it was too large. This behavior can be adjusted by setting
'spark.debug.maxToStringFields' in SparkEnv.conf.
17/11/30 13:43:12 WARN CosmosDBConnection: CosmosDBConnection::Input preferred region list: Central India;
+---------+---------+-------+----------------+-----------+-----------------------+----------------------------+--------+------------+------
-----+---------------+-----------+------------+------------+-----------------------+-----------------------------------------------------------
-------+---------+--------+---------+---------+------------+---------+-----------+------------+---------+--------------+---------+----------
----+---------------+---------+------------+-------------------------------------+-----------------------+-------------------------------
----------------------------+----------+----+-----------------------------------+----+
|AlarmWord|AssetCode|AssetId|AssetSubCategory|CompanyCode|EventEnqueuedUtcTime    |EventProcessedUtcTime       |FuelType|GPSTimeStamp|Gatewa
     |HeightUnderHook|IMEI       |InclinationX|InclinationY|InstanceOccuringDateTime|IoTHub
       |JibLength|Latitude|LoadValue|Longitude|MovementType|NoofFalls|PartitionId|RTCTimeStamp|RatedLoad|ServerDateTime|SlewAngle|SoftwareVe
sion|TrolleyPosition|WindSpeed|_attachments|_etag                                |_rid                    |_self
                            |_ts       |age |id                                  |name|
+---------+---------+-------+----------------+-----------+-----------------------+----------------------------+--------+------------+------
-----+---------------+-----------+------------+------------+-----------------------+-----------------------------------------------------------
-------+---------+--------+---------+---------+------------+---------+-----------+------------+---------+--------------+---------+----------
----+---------------+---------+------------+-------------------------------------+-----------------------+-------------------------------
----------------------------+----------+----+-----------------------------------+----+
|0        |3450031  |3450031|TC              |1          |2017-11-24T08:55:53.779Z|2017-11-24T08:58:11.0869452Z|A       |35:34.0     |RMT285
42153|100            |8.68326E+14|0           |0           |00:04.0                 |[d1,null,null,636471077750159279,2017-11-24T08:55:55.71
Z,null]|5000     |12.93342|280      |77.73848 |S           |2        |2          |00:04.0     |3000     |24:03.0       |149      |V2.04
```

Writing Data to COSMOSDB

1) First upload sample json file with some dummy record to blob storage. These records will be inserted in COSMOSDB



2) Run following command

val persondf = spark.read.json("/example/Person.json")

```
scala> val persondf = spark.read.json("/example/Person.json")
persondf: org.apache.spark.sql.DataFrame = [age: bigint, name: string]
```

3) Run following command to point Azure cosmosdb (where data needs to be written)

val writeConfig = Config(Map("Endpoint" ->
"https://sudhirawdemocdb.documents.azure.com:443/",
"Masterkey" ->
"MstsCKBMXIH8zGfWx4YEhCUcTvBVor9yDYADpg7G393oqYHQX7erWFKE353cWsuoqw3m9nZnx
yjtvQ73lIpmeg==",
"Database" -> "startcosmos",
"PreferredRegions" -> "Central India;",
"Collection" -> "assestdata",
"WritingBatchSize" -> "100"))

```
scala> val writeConfig = Config(Map("Endpoint" -> "https://sudhirawdemocdb.documents.azure.com:443/",
     |       "Masterkey" -> "MstsCKBMXIH8zGfWx4YEhCUcTvBVor9yDYADpg7G393oqYHQX7erWFKE353cWsuoqw3m9nZnxyjtvQ73lIpmeg==",
     |       "Database" -> "\nstartcosmos",
     |       "PreferredRegions" -> "Central India;",
     |       "Collection" -> "assestdata",
     |       "WritingBatchSize" -> "100"))
writeConfig: com.microsoft.azure.cosmosdb.spark.config.Config = com.microsoft.azure.cosmosdb.spark.config.ConfigBuilder$$anon$1@2446dd1a
```

4) Run following command to Upsert dataframe

import org.apache.spark.sql.SaveMode
persondf.write.mode(SaveMode.Overwrite).cosmosDB(writeConfig)

```
scala> import org.apache.spark.sql.SaveMode
import org.apache.spark.sql.SaveMode

scala> persondf.write.mode(SaveMode.Overwrite).cosmosDB(writeConfig)
17/11/30 13:56:38 WARN CosmosDBConnection: CosmosDBConnection::Input preferred region list: Central India;
```

5) Let's read collection to make sure if data got inserted. From Azure portal