

Product Metrics Hive Table

Below tables are created for testing in our bucket

External hive table:

partitioned by cluster,service_name,endpoint_name,date

```
spark-shell --master yarn --num-executors 20 --executor-cores 5 --executor-memory 20G --conf spark.dynamicAllocation.enabled=false --driver-memory 12g --conf spark.ui.filters="" --files "gs://cperf_dev_artifacts/configs/log4j.properties,gs://cperf_dev_artifacts/certs/kafka_ssl_truststore.jks,gs://cperf_dev_artifacts/certs/kafka_ssl_keystore.jks" --conf spark.driver.extraJavaOptions="-Dlog4j.configuration=log4j.properties" --packages "io.delta:delta-spark_2.12:3.2.0,org.apache.hive.hcatalog:hive-hcatalog-core:3.1.3" --conf "spark.sql.extensions=io.delta.sql.DeltaSparkSessionExtension" --conf "spark.sql.catalog.spark_catalog=org.apache.spark.sql.delta.catalog.DeltaCatalog"
sql("SET hive.exec.dynamic.partition=true;").show()
sql("SET hive.exec.dynamic.partition.mode=nonstrict;").show()
sql("set mapreduce.input.fileinputformat.input.dir.recursive=true").show()
sql("set spark.hive.mapred.supports.subdirectories=true").show()
sql("SET hive.json.schema.inference = true").show()

spark.sql(s"add jar /u/users/s0s0j3p//files/json-serde-1.3.8-jar-with-dependencies.jar").show()

sql("""
CREATE EXTERNAL TABLE ww_chnl_perf_stg_qa.nrt_application_logs_test1(
    source_request_id STRING,
    trace_Id STRING,
    response_ts BIGINT,
    method STRING,
    response_body STRING,
    response_bytes INT,
    api_version STRING,
    request_size_bytes INT,
    error_reason STRING,
    request_body STRING,
    created_ts BIGINT,
    supplier_company STRING,
    request_ts BIGINT,
    header STRING,
    endpoint_path STRING
)
PARTITIONED BY (
    cluster string,
    service_name string,
    endpoint_name string,
    date STRING)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
STORED AS TEXTFILE
LOCATION 'gs://ei-nrt-streaming/logs_v2';
""").show()

sql("msck repair table ww_dv_platform_log_stg_qa.api_logs").show()
```

Note: This refreshes newly added partitions only after we trigger msck repair command

External hive table 2:

partitioned by cluster,service_name,endpoint_name

```

sql("""
CREATE EXTERNAL TABLE ww_chnl_perf_stg_qa.nrt_application_logs_test2(
    source_request_id STRING,
    trace_id STRING,
    response_ts BIGINT,
    method STRING,
    response_body STRING,
    response_bytes INT,
    api_version STRING,
    request_size_bytes INT,
    error_reason STRING,
    request_body STRING,
    created_ts BIGINT,
    supplier_company STRING,
    request_ts BIGINT,
    header STRING,
    endpoint_path STRING
)
PARTITIONED BY (
cluster string,
service_name string,
endpoint_name string)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
STORED AS TEXTFILE
LOCATION 'gs://ei-nrt-streaming/logs_v2';
""").show()

sql("msck repair table ww_dv_platform_log_stg_qa.api_logs").show()

```

Note: Refresh is required only if there is new cluster,service_name,endpoint_name . Date will not be available . if we need date to be in table, we can add date in the file

Non partitioned table:

```

sql("""
CREATE EXTERNAL TABLE ww_chnl_perf_stg_qa.nrt_application_logs_test1_new(
    source_request_id STRING,
    trace_id STRING,
    response_ts BIGINT,
    method STRING,
    response_body STRING,
    response_bytes INT,
    api_version STRING,
    request_size_bytes INT,
    error_reason STRING,
    request_body STRING,
    created_ts BIGINT,
    supplier_company STRING,
    request_ts BIGINT,
    header STRING,
    endpoint_path STRING,
    service_name string,
    endpoint_name string
)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
STORED AS TEXTFILE
LOCATION 'gs://ei-nrt-streaming/logs_v2';
""").show()

```

Note: This don't need refresh and it gets updated automatically.

DDL shared with core services team:

```

sql("SET hive.exec.dynamic.partition=true;").show()
sql("SET hive.exec.dynamic.partition.mode=nonstrict;").show()
sql("set mapreduce.input.fileinputformat.input.dir.recursive=true").show()
sql("set spark.hive.mapred.supports.subdirectories=true").show()
spark.sql(s"add jar /u/users/s0s0j3p//files/json-serde-1.3.8-jar-with-dependencies.jar").show()

```

```
sql("""  
CREATE EXTERNAL TABLE ww_dv_audit_log_qa.api_logs(  
    source_request_id STRING,  
    trace_id STRING,  
    response_ts BIGINT,  
    method STRING,  
    response_body STRING,  
    response_size_bytes INT,  
    api_version STRING,  
    request_size_bytes INT,  
    error_reason STRING,  
    request_body STRING,  
    created_ts BIGINT,  
    supplier_company STRING,  
    request_ts BIGINT,  
    header STRING,  
    endpoint_path STRING  
)  
PARTITIONED BY (  
    cluster string,  
    service_name string,  
    endpoint_name string,  
    date STRING)  
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'  
STORED AS TEXTFILE  
LOCATION 'gs://a098b07a-9ae2-4261-82a0-93ac05930312/logs';  
""").show()
```