Kylin Lambda 验证

# 数据准备

设定数据时间跨度为：2020-07-27，2020-07-28，2020-07-29，2020-07-30

每天每小时发送一条数据，即每天 24 条，总共：96 条数据。

## 样例数据

{"id":"20200730181000","ts":1596103800000,"type":1,"counter":1}

## Hive原始表

样例数据对应的Hive原始表。

create table ods\_kylin\_data (

id string,

ts bigint,

type int,

counter int

) COMMENT '原始数据'

STORED AS ORC

TBLPROPERTIES ('parquet.compression'='SNAPPY');

## Lambda Hive Talbe

Kylin Lambda 对应的Hive表

create table kylin\_lambda\_verify (

id string,

ts bigint,

type int,

counter int,

year\_start date,

month\_start date,

hour\_start timestamp,

minute\_start timestamp

) COMMENT 'Lambda Hive Table'

PARTITIONED BY (day\_start date)

STORED AS ORC

TBLPROPERTIES ('parquet.compression'='SNAPPY');

## 数据ETL

INSERT OVERWRITE TABLE kylin\_lambda\_verify PARTITION(day\_start)

SELECT id,

ts,

type,

counter,

FROM\_UNIXTIME(CAST(ts / 1000 AS BIGINT), 'yyyy-01-01') AS year\_start,

FROM\_UNIXTIME(CAST(ts / 1000 AS BIGINT), 'yyyy-MM-01') AS month\_start,

FROM\_UNIXTIME(CAST(ts / 1000 AS BIGINT), 'yyyy-MM-dd HH:00:00') AS hour\_start,

FROM\_UNIXTIME(CAST(ts / 1000 AS BIGINT), 'yyyy-MM-dd HH:mm:00') AS minute\_start,

FROM\_UNIXTIME(CAST(ts / 1000 AS BIGINT), 'yyyy-MM-dd') AS day\_start

FROM ods\_kylin\_data

# Kylin

**时区配置**

kylin.web.timezone=GMT+8

kylin.stream.event.timezone=GMT+8

## Streming Table

图片包含 截图, 游戏机

描述已自动生成

## Kylin Model

### Model Info

图片包含 游戏机, 截图

描述已自动生成

### Data Model

图片包含 游戏机, 截图

描述已自动生成

### Dimensions

图片包含 游戏机, 截图

描述已自动生成

### Measures

图片包含 游戏机, 截图

描述已自动生成

### Settings

图片包含 游戏机, 截图

描述已自动生成

## Kylin Cube

### Cube Info

图片包含 游戏机, 截图

描述已自动生成

### Dimensions

图片包含 游戏机, 截图

描述已自动生成

### Measures

图片包含 游戏机, 截图

描述已自动生成

### Refresh Setting

图片包含 游戏机, 截图

描述已自动生成

### Advanced Setting

图片包含 游戏机, 截图

描述已自动生成

### Configuration Overwrites

图片包含 游戏机, 截图

描述已自动生成

### Overview

图片包含 游戏机, 截图

描述已自动生成

# 验证

构建历史

图片包含 游戏机, 截图

描述已自动生成

将对 2020-07-27数据进行刷新

## 查询条数

每小一条数据，查询结果正确

图片包含 游戏机, 截图

描述已自动生成

## 查询 07-28 日数据时间

图片包含 游戏机, 截图

描述已自动生成

## 刷新 07-27 日数据

图片包含 游戏机, 截图

描述已自动生成

## 刷新27日数据后查询

### 统计每日数据条数

图片包含 游戏机, 截图

描述已自动生成

### 查询 2020-07-27 数据

图片包含 游戏机, 截图

描述已自动生成

### 查询 2020-07-28 数据

图片包含 游戏机, 截图

描述已自动生成

# 附录

## Hive原始表数据

INSERT INTO ods\_kylin\_data VALUES ('20200727001000', 1595779800000, 1, 1),('20200727011000', 1595783400000, 1, 1),('20200727021000', 1595787000000, 1, 1),('20200727031000', 1595790600000, 1, 1),('20200727041000', 1595794200000, 0, 1),('20200727051000', 1595797800000, 0, 1),('20200727061000', 1595801400000, 0, 1),('20200727071000', 1595805000000, 0, 1),('20200727081000', 1595808600000, 0, 1),('20200727091000', 1595812200000, 0, 1),('20200727101000', 1595815800000, 0, 1),('20200727111000', 1595819400000, 0, 1),('20200727121000', 1595823000000, 0, 1),('20200727131000', 1595826600000, 1, 1),('20200727141000', 1595830200000, 1, 1),('20200727151000', 1595833800000, 1, 1),('20200727161000', 1595837400000, 0, 1),('20200727171000', 1595841000000, 1, 1),('20200727181000', 1595844600000, 1, 1),('20200727191000', 1595848200000, 0, 1),('20200727201000', 1595851800000, 0, 1),('20200727211000', 1595855400000, 1, 1),('20200727221000', 1595859000000, 0, 1),('20200727231000', 1595862600000, 1, 1),('20200728001000', 1595866200000, 0, 1),('20200728011000', 1595869800000, 1, 1),('20200728021000', 1595873400000, 0, 1),('20200728031000', 1595877000000, 1, 1),('20200728041000', 1595880600000, 1, 1),('20200728051000', 1595884200000, 1, 1),('20200728061000', 1595887800000, 1, 1),('20200728071000', 1595891400000, 0, 1),('20200728081000', 1595895000000, 0, 1),('20200728091000', 1595898600000, 1, 1),('20200728101000', 1595902200000, 1, 1),('20200728111000', 1595905800000, 0, 1),('20200728121000', 1595909400000, 1, 1),('20200728131000', 1595913000000, 0, 1),('20200728141000', 1595916600000, 0, 1),('20200728151000', 1595920200000, 0, 1),('20200728161000', 1595923800000, 1, 1),('20200728171000', 1595927400000, 1, 1),('20200728181000', 1595931000000, 1, 1),('20200728191000', 1595934600000, 1, 1),('20200728201000', 1595938200000, 0, 1),('20200728211000', 1595941800000, 0, 1),('20200728221000', 1595945400000, 1, 1),('20200728231000', 1595949000000, 1, 1),('20200729001000', 1595952600000, 0, 1),('20200729011000', 1595956200000, 0, 1),('20200729021000', 1595959800000, 1, 1),('20200729031000', 1595963400000, 1, 1),('20200729041000', 1595967000000, 0, 1),('20200729051000', 1595970600000, 1, 1),('20200729061000', 1595974200000, 0, 1),('20200729071000', 1595977800000, 1, 1),('20200729081000', 1595981400000, 0, 1),('20200729091000', 1595985000000, 0, 1),('20200729101000', 1595988600000, 1, 1),('20200729111000', 1595992200000, 1, 1),('20200729121000', 1595995800000, 0, 1),('20200729131000', 1595999400000, 1, 1),('20200729141000', 1596003000000, 1, 1),('20200729151000', 1596006600000, 1, 1),('20200729161000', 1596010200000, 0, 1),('20200729171000', 1596013800000, 1, 1),('20200729181000', 1596017400000, 1, 1),('20200729191000', 1596021000000, 1, 1),('20200729201000', 1596024600000, 0, 1),('20200729211000', 1596028200000, 1, 1),('20200729221000', 1596031800000, 0, 1),('20200729231000', 1596035400000, 1, 1),('20200730001000', 1596039000000, 1, 1),('20200730011000', 1596042600000, 0, 1),('20200730021000', 1596046200000, 1, 1),('20200730031000', 1596049800000, 1, 1),('20200730041000', 1596053400000, 0, 1),('20200730051000', 1596057000000, 1, 1),('20200730061000', 1596060600000, 1, 1),('20200730071000', 1596064200000, 0, 1),('20200730081000', 1596067800000, 1, 1),('20200730091000', 1596071400000, 0, 1),('20200730101000', 1596075000000, 1, 1),('20200730111000', 1596078600000, 1, 1),('20200730121000', 1596082200000, 0, 1),('20200730131000', 1596085800000, 1, 1),('20200730141000', 1596089400000, 1, 1),('20200730151000', 1596093000000, 1, 1),('20200730161000', 1596096600000, 1, 1),('20200730171000', 1596100200000, 1, 1),('20200730181000', 1596103800000, 1, 1),('20200730191000', 1596107400000, 0, 1),('20200730201000', 1596111000000, 0, 1),('20200730211000', 1596114600000, 0, 1),('20200730221000', 1596118200000, 0, 1),('20200730231000', 1596121800000, 0, 1)

## 发送数据到Kafka

public static void main(String[] args) throws Exception {

Random random = new Random();

Properties props = new Properties();

props.put(ProducerConfig.BOOTSTRAP\_SERVERS\_CONFIG, bootstrapServers);

props.put(ProducerConfig.KEY\_SERIALIZER\_CLASS\_CONFIG, "org.apache.kafka.common.serialization.StringSerializer");

props.put(ProducerConfig.VALUE\_SERIALIZER\_CLASS\_CONFIG, "org.apache.kafka.common.serialization.StringSerializer");

KafkaProducer<String, String> producer = new KafkaProducer<>(props);

List<String> days = Lists.newArrayList("2020-07-27", "2020-07-28", "2020-07-29", "2020-07-30");

List<String> hours = Lists.newArrayList();

for (int i = 0; i < 24; i++) {

hours.add(StringUtils.leftPad(String.valueOf(i), 2, "0"));

}

StringBuilder sb = new StringBuilder("INSERT INTO ods\_kylin\_data VALUES ");

for (String day : days) {

for (String hour : hours) {

String s = day + " " + hour + ":" + "10:00";

Date date = DateUtils.parseDate(s, "yyyy-MM-dd HH:mm:ss");

String id = DateFormatUtils.format(date, "yyyyMMddHHmmss");

long ts = date.getTime();

int type = random.nextInt(2);

int counter = 1;

MyData myData = new MyData(id, ts, type, counter);

String json = JSON.toJSONString(myData);

ProducerRecord<String, String> record = new ProducerRecord<>("ods\_kylin\_data", json);

producer.send(record, (rm, e) -> {

if (Objects.nonNull(e)) {

System.out.println(e.getMessage());

} else {

System.out.println("topic: " + rm.topic() + ", partition: " + rm.partition() + ", offset: " + rm.offset());

}

});

// 插入到Hive的SQL

sb.append("('").append(id).append("', ")

.append(ts).append(", ")

.append(type).append(", ")

.append(counter).append("),");

}

}

String sql = sb.deleteCharAt(sb.length() - 1).toString();

System.out.println(sql);

producer.close();

}

static class MyData {

private String id;

private Long ts;

private Integer type;

private Integer counter;

public MyData() {

}

public MyData(String id, Long ts, Integer type, Integer counter) {

this.id = id;

this.ts = ts;

this.type = type;

this.counter = counter;

}